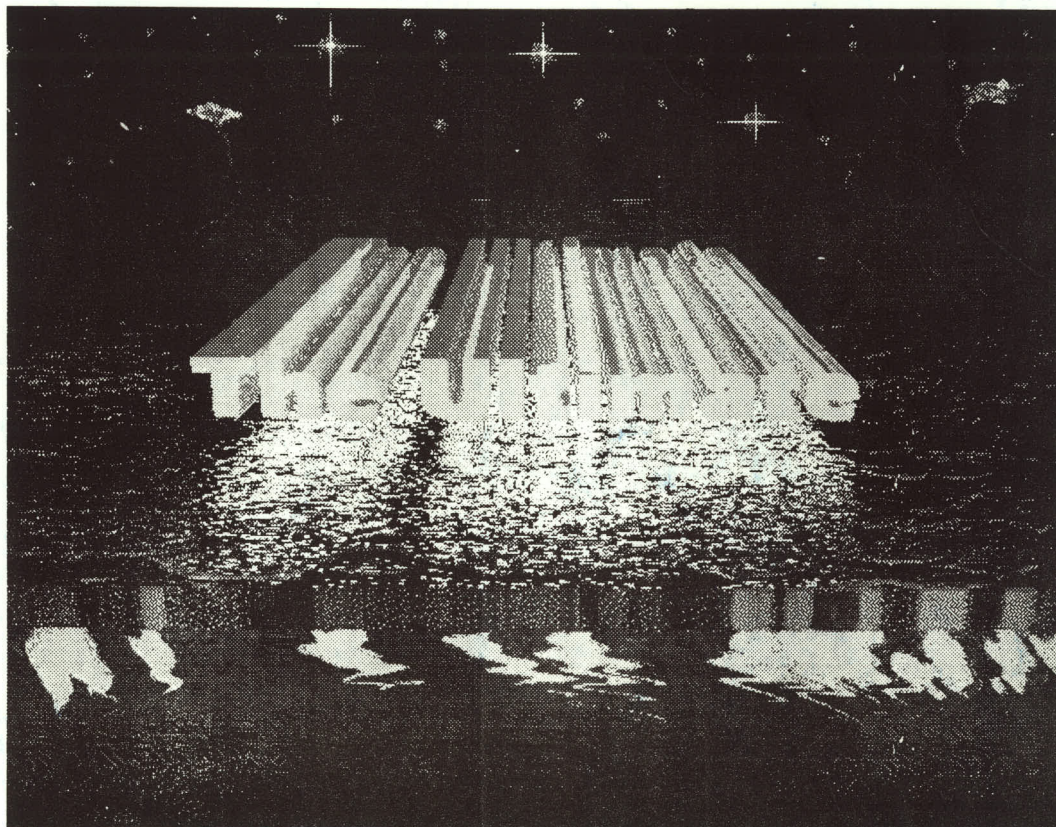




De μ P Kenner

Zestiende jaargang nr. 3
Augustus 1992
77



In dit nummer o.a.:

Trace voor 680X0

Chaos rond de 80486

Hoe schrijf ik een artikel?

Voortgang KGN-68k werkgroep

Inleiding rond de 68030 en supportchips

Enschede, 5 oktober 1992

Beste leden,

Deze μ P Kenner is niet alleen wat aan de dunne kant, maar hij is ook rijkelijk laat. Daarvoor onze excuses. Voor het (veel te) laat uitkomen van uw clubblad is echter een reden. Die reden is... deze uitnodiging. Jawel, een uitnodiging! Voor een extra clubbijeenkomst. Deze clubbijeenkomst zal geheel gewijd zijn aan het thema "Linux". Dat is, zoals u misschien al wel weet, een public domain UNIX voor de 386 en hoger. Er wordt overwogen in een later stadium deze UNIX kloon te porten naar de KGN68k/30.

Terug echter naar het onderwerp van dit schrijven. Er is dus een extra bijeenkomst, deze keer in Utrecht. het programma en een routebeschrijving voor het openbaar vervoer vindt u hieronder.

Namens het bestuur en de redactie,

Joost Voorhaar.

Uitnodiging clubbijeenkomst

Datum : 24 oktober 1992
Tijd : Vanaf 10:00 uur
Locatie : PMWZ-Oost,
Bouwstraat 55
3572 SP Utrecht
Thema : Linux
Entree : Gratis (ook niet-leden zijn van harte welkom!)

Naast verschillende systemen hebben we o.a. gezorgd voor een aantal speciale hardware aanbiedingen en wellicht nog meer UNIX-specifieke attracties.

Routebeschrijving

Het gebouw is per bus bereikbaar vanaf het centraal station met de lijnen 4 en 11. Neemt u lijn 4, dan moet u uitstappen bij de Biltstraat, voor lijn 11 stapt u uit bij de halte Kruisstraat.

In verband met de slechte parkeermogelijkheden in de directe omgeving van het gebouw raden wij u sterk aan gebruik te maken van het openbaar vervoer! Indien u echter uw systeem mee wilt nemen en dus met de auto komt: helaas, een goede routebeschrijving is nog niet beschikbaar. Om u een beetje op weg te helpen: volgens het 100.000 stratenboek van Shell is de Bouwstraat een zijstraat van de Obrechtstraat. En dat is weer een zijstraat van de beroemdste straat van het Monopoly: de Biltstraat. Zodra er meer informatie beschikbaar is zullen wij dit op het bulletin board bekend maken.

Inhoudsopgave

De μ P Kenner

Nummer 77, augustus 1992

Verschijnt 6 maal per jaar

Oplage: 250 stuks

Druk: FEBO Offset, Enschede

De redactie:

Joost Voorhaar

Gert van Opbroek

Geert Stappers

Nico de Vries

Eindredactie:

Joost Voorhaar

Vormgeving:

Joost Voorhaar

Nico de Vries

Redactieadres:

Joost Voorhaar

Jekerstraat 67

7523 VP Enschede

De μ P Kenner nummer 78 verschijnt op
17 oktober 1992.

Kopijsluitingsdatum voor nummer 78 is
vastgesteld op 3 oktober 1992.

Vereniging

Uitnodiging voor de clubbijeenkomst	5
Credit card beschikbaar voor KGN leden	8
Voortgang KGN68k	33
We hebben een ridder in ons midden	35
Van de bestuurstafel	37

Algemeen

Redactioneel	4
Wintertime	7
Hoe schrijf je een artikel voor de μ P Kenner?	9
Naschrift van de redactie	12
Te Koop: DOS65 Computer + toebehoren	12
Nieuwtjes en andere wetenswaardigheden	30
Connect 2400	31
Altijd de juiste tijd	34
Gids voor electro-technici	36

Software/Talen

To Share Or Not To Share, That's The Question... ..	6
Trace exception routine voor MC68000	13
Registratie van shareware	26

Systemen

De Motorola 68030; het hart van KGN68k	19
----------------------------------------------	----

Hardware

De ene 486 is de andere niet... ..	27
KGN68k ISA busprintdefinitie	35

De μ P Kenner is het huisorgaan van de KIM gebruikersclub Nederland en wordt bij verschijnen gratis toegezonden aan alle leden van de club. De μ P Kenner verschijnt vijf maal per jaar, in principe op de derde zaterdag van de maanden februari, april, augustus, oktober en december.

Kopij voor het blad dient bij voorkeur van de leden afkomstig te zijn. Deze kopij kan op papier, maar liever in machine-leesbare vorm opgestuurd worden aan het redactieadres. Kopij kan ook op het Bulletin Board van de vereniging gepost worden in de redactie area. Nadere informatie kan bij het redactieadres of via het bulletin board opgevraagd worden.

De redactie houdt zich het recht voor kopij zonder voorafgaand bericht niet of slechts gedeeltelijk te plaatsen of te wijzigen. Geplaatste artikelen blijven het eigendom van de auteur en mogen niet zonder diens voorafgaande schriftelijke toestemming door derden gepubliceerd worden, in welke vorm dan ook.

De redactie noch het bestuur kan verantwoordelijk gesteld worden voor toepassing(en) van de geplaatste kopij.

Redactioneel

Het zomerseizoen is achter de rug. En dat zullen we weten ook... wat een regen is er gevallen de afgelopen dagen zeg! Een compleet zwembad kun je er mee vullen lijkt het wel.

Maar goed, deze dagen zijn niet alleen goed voor de tuin, maar ook voor de KGN. Het stof wordt weer van de monitor geveegd, de toetsenborden beginnen weer te rammelen en een enkele soldeerbout wordt weer warmgestookt. Kortom: tijd voor het winterseizoen.

Uw redacteur kan het winterseizoen kleurig ingaan. Inmiddels is mijn persoonlijke systeem namelijk uitgebreid met een echte super-VGA kleurenmonitor, er is wat extra geheugen in gekomen en een heuse coprocessor heeft het lege voetje naast de processor opgevuld. Kortom: power genoeg om uw kopij op een luxe manier te verwerken...

En in dat laatste zit 'm de kneep. We hebben nog altijd een zwaar tekort aan schrijvers. Om aan dit probleem een eind te maken wil ik de structuur van de redactie ingrijpend gaan wijzigen. Tot nog toe was er eigenlijk alleen een hoofdredacteur die teksten en tekeningen verzamelde en een Nico die van het bijeengeraapte zootje altijd weer een keurige μ P Kenner weet te maken. Gelukkig zijn er wel een aantal personen die iedere keer zorgen voor een hoop interessant kopij, maar ook deze groep personen raakt een keer leeggeschreven.

De bedoeling is een min of meer "echte" redactie op poten te zetten. Deze redactie zal gaan bestaan uit een vaste groep schrijvers die op projectbasis kopij

maken en verwerken. Doordat er meer in groepsverband gewerkt kan worden is het geen enkel probleem als één of meer van de redactieleden zichzelf niet in staat acht om leesbare artikelen af te leveren. In de eerste plaats kan op deze manier iedereen leren hoe je een artikel kunt bouwen (zie ook het artikel van Gert van Opbroek hierover), maar zeker niet in de laatste plaats kan zo iemand ook uitstekend materiaal verzamelen, sorteren en uitpluizen. Schrijven is namelijk veel meer dan alleen een leesbaar verhaal maken... je zult een stukje onderzoek moeten doen naar datgene waarover je wilt schrijven en controleren of de kennis die je van het betreffende onderwerp hebt accuraat is. Uiteraard gaat is dit onderzoek minder belangrijk voor de bespreking van een zelfgebouwd stuk hardware of een zelfgeschreven programma.

Deze keer zoeken we dus niet alleen mensen die met de pen overweg kunnen, maar ook zij die zich op projectbasis willen verdiepen in alles dat met computertechniek te maken heeft worden verzocht contact op te nemen met de redactie. De μ P Kenner, het BBS en alle andere activiteiten van de KGN kunnen alleen dan overleven als er ook leden zijn die zich willen en kunnen inzetten voor de vereniging. En waarom dan niet als "speurneus" voor de redactie?

Al met al zijn we er deze keer toch weer in geslaagd om een μ P Kenner te maken. Minder dik dan u wellicht van ons gewend bent, maar de redenen daarvoor heeft u wellicht al begrepen...

Joost Voorhaar

Uitnodiging voor de clubbijeenkomst

Datum: 19 september 1992
 Locatie: Wijkcentrum 't Veurbrook
 Jan Tooropstraat 27
 7606 Almelo
 Telefoon: 05490 - 10353
 Entree: gratis
 Thema: Softwareontwikkeling voor de KGN-68k

Routebeschrijving

Per auto, vanuit het westen en het zuiden (A1/A35):

1. Aan het einde van de snelweg rechtsaf. Bij het eerstvolgende kruispunt **MET VERKEERSLICHTEN** linksaf, richting Wierden/Zwolle. Bij de eerstvolgende verkeerslichten recht door. Bij de volgende verkeerslichten (links BP tankstation en Opel garage Kamp) gaat u rechtsaf.

2. U rijdt nu op de Windmolenbroeksweg. Doorrijden tot over de brug, dan de eerste straat rechts. Dit is de W. van Konijnenburgstraat. Na plm. 50 meter rechtsaf. Dit is de Tooropstraat. Met de bocht mee naar links. Na plm. 50 meter aan de rechterkant: 't Veurbrook.

Per auto, vanuit het noorden (via de N 36):

1. Bij de stoplichten rechtsaf, richting streekziekenhuis. U bevindt zich nu op de rondweg om Almelo. Deze weg blijven volgen tot u het BP tankstation ziet bij dit kruispunt linksaf. Zie verder punt 2.

Met openbaar vervoer:

Vanaf NS-station Almelo met de stadsbus naar de wijk Molenbroek. Uitstappen bij de halte Windmolenbroeksweg. Schuin tegenover de bushalte staat een wegwijzer, daarop staat ook 't Veurbrook vermeld.

Programma:

09:30 Zaal open met koffie
 10:15 Opening
 10:30 Spreekbeurt van Geert Stappers over de softwareontwikkeling voor de KGN-68k.
 11:45 Forum en markt
 12:00 Lunch, consumpties tegen betaling
 Aansluitend het informele gedeelte met de mogelijkheid om andermans systemen te bewonderen en Public Domain software uit te wisselen. U en uw systeem zijn uiteraard van harte welkom.

17:00 Sluiting

Let op:

Het is ten strengste verboden illegale kopieën van software te verspreiden. Aan personen die deze regel overtreden zal de verdere toegang tot de bijeenkomst ontzegd worden. Breng verder alleen software mee die u legaal in uw bezit heeft. Het bestuur aanvaardt geen enkele aansprakelijkheid voor de gevolgen van het in bezit hebben van illegale software.



To Share Or Not To Share, That's The Question...

Als u regelmatig op machineniveau programmeert, dan kent u het probleem: de documentatie van uw computersysteem is veelal niet compleet of staat verspreid over een groot aantal publicaties. Wie veel met MS-DOS programmeert kent waarschijnlijk Ralf Brown's interrupt lijst wel, en wellicht bent u dan ook bekend met "Tech Help!" van Flambeaux software (een commercieel programma overigens). Een veel flexibeler (shareware-) variant hierop is HelpPC van David Jurgens.

HelpPC lijkt wel erg veel op Tech Help!. Een paar verschillen zijn er natuurlijk wel. Zo zijn de databestanden van HelpPC plain ASCII zodat je zelf makkelijk nieuwe manuals kunt toevoegen, bestaande manuals kunt aanpassen of helemaal verwijderen. Verder bestaat het programma uit een drietal losse programma's (HelpPC.Com, HelpTSR.Com en IndexHlp.Com), terwijl Tech Help! één enkele EXE-file is (wel aangevuld met de manualfiles natuurlijk). Verder lijken de programma's wat werking betreft als twee druppels water op elkaar. Beide kunnen ze, als ze resident geladen zijn, context sensitive hulp bieden. Dat wil zeggen dat gekeken wordt naar het woord waar de cursor zich op dat moment op bevindt. Het help-systeem zoekt dan naar dat specifieke woord in zijn index.

Documentatie

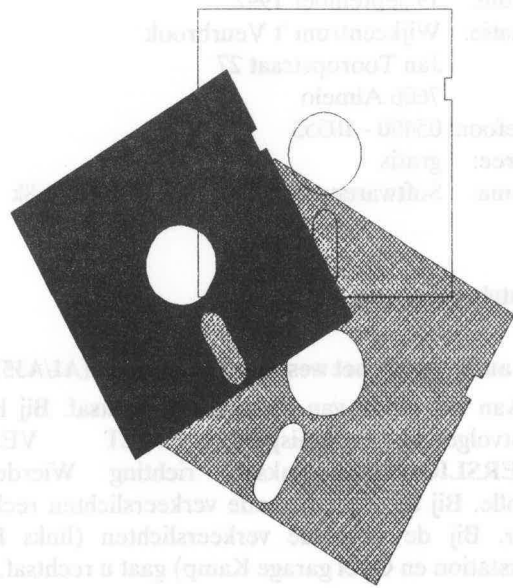
Door de eenvoud van HelpPC is er niet zo heel veel documentatie nodig. Dat blijkt alleen al uit de omvang: slechts tien pagina's die in totaal zo'n 26 kB aan informatie bevatten. De documentatie is compact maar toch helder. Wel ontbreekt een paginanummering en is er geen index en/of inhoudsopgave. Door de geringe omvang van het bestandje mag dit echter niet zo'n probleem vormen.

Systeemeisen

"Small is beautiful" is niet bepaald een kenmerk van de moderne programmatuur. Een aangename uitzondering hierop vormt HelpPC, want het programma is niet alleen compact, maar ook de eisen die HelpPC aan het systeem stelt zijn minimaal. Een IBM-DOS compatible systeem waarop tenminste MS- of PC-DOS 2.0 draait en 64 kB aan RAM is alles. De TSR versie neemt 32 kByte in beslag, aangevuld met de ruimte die nodig is voor de indexfiles. Een harde schijf is niet persé noodzakelijk, maar wordt wel sterk aangeraden.

Features

Ondanks de relatief geringe omvang heeft HelpPC toch een aantal hele aardige features; de meeste hiervan hebben betrekking op de TSR-versie. Zo



maakt HelpPC gebruik van de "alternate monitor" in een dual-monitor systeem. De tekst komt dan op de niet-actieve monitor te staan zodat deze het normale beeld niet overschrijft. Verder is het ding netwerk compatible, wat veel van dit soort programmaatjes nog wel eens missen. Het ding kan zichzelf uit het geheugen verwijderen, ook als 'ie in high memory geladen is. Tenslotte heeft de auteur voorzien in een integrity check waarmee de TSR zichzelf eerst controleert alvorens op te starten. Hierdoor voorkomt hij systeem crashes die voortkomen uit andere programmatuur die (geheel ten onrechte) geheugen gebruiken die door HelpPC in beslag genomen is. Als het trigger- en checking mechanisme overschreven is kan HelpPC er (uiteeraard) ook niets meer aan doen en zal de machine hoogstwaarschijnlijk crashen (lees: al gecrasht zijn...). In de praktijk komen dit soort problemen natuurlijk niet zo vaak voor, maar het is een aardig detail.

Manuals

Standaard worden er een aantal aardige manuals meegeleverd. Deze manuals hebben als "main topic": assembler, C, hardware, interrupts, DOS- en BIOS tabellen en tenslotte is er een algemene vergaarbak. Volgens de handleiding zou er ook een manual meegeleverd moeten zijn met DOS commando's, maar die ontbreekt in alle archives die ik gezien heb. De assembler guide is bijgewerkt tot en met de '86 en gaat uit van MicroSoft's MASM. De C-guide gaat uit van MicroSoft C. Helaas, want er zijn geen aanwijzingen opgenomen ten aanzien van compatibiliteit met bijvoorbeeld ANSI-C. Een (C++) update zou hier wel op z'n plaats zijn. Onder het kopje "hardware data and specifications"

bevindt zich een mengelmoesje van IC-beschrijvingen en IBM-implementatie details. Zelfs als pure hardwareman kun je profijt hebben van deze handleiding, want ook de signalen van de PC-bus zijn omschreven. Een timingdiagram ontbreekt echter.

De "interrupt guide" houdt zich voornamelijk bezig met de DOS function dispatcher, maar ook de BIOS en muisinterrupts komen aan bod.

Zelf maken

Zoals gezegd kun je zelf handleidingen aanmaken en wijzigen. Als je hier even mee bezig bent komen er toch wat slechte eigenschappen van het systeem naar boven. Een handleiding bestaat uit een normale ASCII tekstfile, waarvan de eerste kolom gereserveerd is voor stuurtekens. Deze stuurtekens zijn ook gewone ASCII-tekens. Zo betekent een dubbele punt in de eerste kolom bijvoorbeeld dat de regel als subtopic gebruikt moet worden. Op de manier waarop één en ander omschreven is in de documentatie zou je mogen verwachten dat er een hele informatieboom gemaakt zou kunnen worden, waarmee je een heel gestructureerde handleiding zou kunnen maken. Dat blijkt echter niet juist te zijn. Er kan een hoofdmenu gemaakt worden waar de afzonderlijke

tekstfiles in opgenomen zijn. Een tekstfile kan verder bestaan uit een aantal opties, waardoor iedere afzonderlijke file gezien kan worden als een submenu. Alle steekwoorden komen in dit submenu voor. Er kan dus geen boomstructuur in de tekstbestanden zelf worden gedefinieerd.

Conclusie

HelpPC is een heel leuk dingetje om op eenvoudige wijze een on-line hulpsysteem in elkaar te draaien. Om op efficiënte wijze informatie terug te vinden zul je echter toch een beter gestructureerd systeem nodig hebben. Op de manier waarop HelpPC de hele handel indeelt kunnen geen overzichtelijke hulpsystemen gebouwd worden. De bijgeleverde manuals voegen echter weer wat aan waarde toe aan het hele systeem. Misschien kunnen we in de toekomst nog wel meer van dit programmaatje verwachten. Nu wekt het geheel een beetje de indruk als ware het een produkt van een ietwat luie programmeur die het allemaal al snel wel best vindt. Met wat meer werk zou het echter zeker een goede vervanger kunnen worden voor de commerciële Norton-guide, THelp (Borland) en TechHelp.

Joost Voorhaar

Wintertime

De winter is weer in aantocht. Nou ja, als je naar het weer buiten kijkt lijkt het inderdaad al herfst... Natuurlijk betekent dat ook weer: tijd voor de HCC-dagen. Die zijn dit jaar op 21 en 22 november gepland. Of wij er dit jaar ook weer als KGN zullen staan is nog even de vraag. De "hoge heren bestuurders" van de HCC vonden het dit jaar niet meer nodig om (zuster-) verenigingen een gereduceerd tarief aan te bieden. Een stand op de HCC-dagen is daardoor wel erg duur geworden; een eenvoudig kraampje moet al f 1100,- opbrengen, een voor de KGN veel te hoge prijs.

Deze prijs hoeft niet betaald te worden door verenigingen die aan de HCC duidelijk kunnen maken dat de aanwezigheid van die vereniging iets toevoegt aan (de waarde van) de beurs. Momenteel worden er dan ook pogingen ondernomen om zo'n (gratis) stand te bemachtigen. De KGN heeft dit jaar namelijk inderdaad iets speciaals te bieden: de presen-

tatie van de KGN68k/30 aan het grote publiek! Vanuit de 68k-werkgroep is toegezegd tot het uiterste te gaan om op de HCC-dagen een werkend prototype te kunnen tonen. Dit prototype zal (uiteraard) nog geen Minix kunnen draaien, maar een "Hello world" en een werkende monitor behoren wel tot de mogelijkheden.

In afwachting van de resultaten van onze inspanningen zullen we doorgaan met de voorbereiding van onze presentatie op de HCC-dagen. Daarvoor proberen we natuurlijk ook weer een gemotiveerde stand-bemanning bij elkaar te schrapen die het gezicht van de vereniging aan de man weet te brengen. Geïnteresseerden kunnen zich natuurlijk altijd melden bij één van de bestuursleden...

Joost Voorhaar

Credit card beschikbaar voor KGN leden

Niet schrikken! Dit is geen advertentie voor de nieuwe "Ikke Njente Geld" bank! Wat dan wel de bedoeling is? C'est simple...

Stel, je wilt ergens op de planeet aarde iets van computerhard- of -software kopen of registreren. Vaak zou het dan vreselijk handig zijn om een credit card te hebben. Een credit card? De KGN heeft er één! En als we toch zo'n ding hebben als club, willen we er ook graag gebruik van maken. En dan bedoelen we dat jullie, de leden, daar van kunnen meegenieten.

Het gebruik

Natuurlijk heb je ook ergens zo'n programmaatje dat je eigenlijk best wel zou willen registreren. Alleen al die rompslomp met giro's, banken en internationale postwissels... terwijl de auteur je ook vaak de mogelijkheid biedt via een credit card te registreren! Als de auteur daarbij vermeldt dat hij MasterCard accepteert kan de KGN je een handje helpen:

- Maak het betreffende bedrag over naar een van de KGN rekeningen, ABN-AMRO Bank 53.43.10.699 of naar de PostBank # 3757649, met een omschrijving waarvoor het bedrag bedoeld is.
- Indien het bedrag naar een buitenlandse rekening moet (wat meestal het geval zal zijn omdat je het anders ook wel via je eigen rekening kunt doen) hanteer je even een grove koers. Een te veel aan gelden overgemaakt bedrag wordt weer teruggestort op je rekening en je krijgt een copie van het afschrift per ommegaande terug. Zo kunnen er geen meningsverschillen ontstaan over te betalen c.q. betaalde bedragen e.d.; dat

is tenslotte het laatste wat we zouden willen!

Houdt een ruime koers aan omdat de bank wel eens provisie zou kunnen rekenen voor het hele gebeuren en voor de betaling van de postzegel voor het betalingsbewijs zodat je weet hoe alles is afgehandeld.

- Indien er een registratieformulier opgestuurd moet worden, stuur die dan even naar ons postbus nummer (postbus 1336 7500 BH Enschede). Dit opdat we ons creditcardnummer op het formulier in kunnen vullen en het formulier kunnen ondertekenen (anders kan er geen geld opgevraagd worden van onze creditcardrekening). Vul alle andere gegevens op het formulier wel even zelf in. En niet te vergeten: vermeld erbij waar het allemaal heen gestuurd moet worden!

De registratie procedure (of aankoop) wordt gestart zodra het geld op de rekening staat en het begeleidende briefwerk is ontvangen. Dat begeleidende briefwerk kan natuurlijk ook via het BBS geregeld worden. De 'boekhouding' is tenslotte op het zelfde adres gevestigd als het BBS... Stuur dan een bericht gericht aan Jacques Banser (da's ikke dus) in de Aan/Van SysOp message area en drop daar ook even het registratie/aankoopformulier.

Indien er nog vragen zijn hoor ik ze graag. Het liefst op het BBS, dan kan ik als dat nodig is het een en ander eerst even uitzoeken, of bel me op, als er erg veel haast achter zit, op nummer 053-324137 liefst tussen 20:00 en 22:00 uur.

Okidokie, dat was het ongeveer wel denk ik,

Groetens, Jacques (jullie penningmeester)

Ik heb interesse in de KGN en wil

☐ Lid worden van de KGN

☐ Meer informatie over de KGN

Naam : _____

Adres : _____

Postcode en Woonplaats : _____

Datum : _____ Handtekening : _____

Dit strookje kunt u ingevuld opsturen aan het secretariaat van: **KIM Gebruikersclub Nederland**
Postbus 1336
7500 BH Enschede

Hoe schrijf je een artikel voor de μ P Kenner?

Inleiding

Waarschijnlijk is het je wel opgevallen dat we binnen de club de laatste tijd wat gebrek aan kopij hebben. Dit is geen probleem van de laatste maanden, nee het is een probleem waar ik als ex-redacteur bijna continue mee geconfronteerd werd.

Nu wil het geval dat er diverse clubleden zijn die mij wel eens aangesproken hebben met de opmerking: "Ik zou wel wat willen schrijven, maar ik denk niet dat ik dat kan" of "Ik weet niet hoe ik het aan moet pakken". Ik zal in dit stukje eens proberen duidelijk te maken hoe ik dat bij de artikelen die ik geschreven heb en nog steeds schrijf aanpak.

"Kan ik het wel?"

Er zijn maar weinig mensen die zeggen "Een artikeltje schrijven? Dat is voor mij een fluitje van een cent". Nee, de meeste mensen hebben grote twijfels. Zo zegt men bijvoorbeeld: "Ja, ik heb wel een leuk programma (of stukje hardware) gemaakt, maar wie zit daar nu op te wachten"? Tja, dat hangt er vanaf. In de meeste gevallen zijn er in de club wel een aantal mensen die zeer geïnteresseerd zijn. Natuurlijk gaat niet iedereen meteen het programma intikken of de hardware nabouwen maar er zitten vaak hele goeie ideeën in die een ander ook weer kan gebruiken. Mijn ervaring is dat er voor vrijwel alles wel enkele aandachtige lezers zijn en dat maakt het, wat bij betreft, al de moeite waard in de μ P Kenner er een artikel(tje) aan te wijden.

Een tweede opmerking die je vaak tegenkomt is: "Ik ben maar een eenvoudige knutselaar en anderen vinden wat ik doe misschien maar simpel". Kijk, dat vindt ik nu het grote voordeel van een hobbyclub. In een hobbyclub vindt je een breed scala aan mensen. Je vindt er zowel de professionals die een opleiding op automatiseringsgebied hebben gevolgd en die ook in de automatisering werkzaam zijn als mensen voor wie de computer alleen maar hobby is. Dit zijn mensen waarvoor ik (als professional) grote bewondering heb. Zij hebben door veel vrije tijd in hun hobby te steken vaak heel veel weten te bereiken en juist die mensen kunnen anderen helpen om verder te komen. Als laatste wil ik hierover nog opmerken

dat de mens van nature erg bescheiden is en dat men vaak twijfelt over zijn eigen kunnen. Stap daarover heen en laat collega-clubleden ook eens zien waar je mee bezig bent. Zelfs professionals kunnen vaak van echte amateurs nog iets leren.

Een derde opmerking die je wel eens hoort is: "Ik zou wel willen, maar ik weet niet hoe ik het aan moet pakken". Welnu, daarover gaat de rest van dit artikeltje.

"Waarover zal ik schrijven?"

Als je voor het eerst een artikel schrijft is het waarschijnlijk het verstandigst te gaan schrijven over iets waar je veel vanaf weet. Verder zou ik in het begin zeker niet een groot en uitgebreid onderwerp uitzoeken. De kans dat je daarin vastloopt is namelijk tamelijk groot en waarom zou je het jezelf moeilijker maken dan nodig is. Nee, kies in het begin iets uit dat je goed kunt overzien. Uiteraard kan dat van alles zijn: de beschrijving van een chipje, een schakelingetje, een programmaatje in taal XYZ dat het getal PI uitrekent in 367 decimalen etc. Ik weet nog wel dat mijn eerste bijdrage aan de μ P Kenner een verhaaltje was over een fout die ik in de FORTH compiler voor de Junior ontdekt had (en had opgelost; 6502 Kenner 34). Ik heb dat gedaan in de vorm van een "Brief aan de redactie". Willem van Pelt, de toenmalige redacteur,

heeft mij vervolgens zo ver weten te krijgen dat ik regelmatig stukjes in het clubblad ging publiceren met als gevolg dat ik het steeds leuker ging vinden en enige jaren redacteur geweest ben.

Ook maken veel kleine artikeltjes van verschillende mensen het blad voor een hoop mensen veel leuker dat een blad dat door twee of drie mensen volgeschreven wordt met lange artikelen en listings. Nee, ik ben van mening dat het blad "Voor elck wat wilsch" moet bevatten en daarvoor zijn korte bijdragen van veel mensen eigenlijk een noodzaak.

Behalve dat je wat kunt schrijven over iets waar je veel vanaf weet, kun je ook iets gaan schrijven over een onderwerp waar je vrijwel niets vanaf weet. Dat is een aanpak die ik de laatste jaren vaak gebruikt

heb. Ik kom op de één of andere manier een onderwerp tegen (netwerken, floating point getallen, harde schijven) waarvan ik denk "Hé, dat is leuk voor de μ P kenner". Vervolgens ga ik op zoek naar informatie over dat onderwerp. Vaak kan ik dat terugvinden in de boeken en tijdschriften die ik in mijn bezit heb en anders bestaan er ook bibliotheken waar je dergelijke informatie kunt opzoeken. Nadat ik het onderwerp bestudeerd heb, schrijf ik op wat ik geleerd heb en dat is dan mijn artikel. Voor mij is dat een hele leuke manier om me in te werken in een bepaald onderwerp en ik krijg de indruk dat een hoop mensen de artikelen die zo ontstaan best aardig vinden. Verder kan ik de kennis die ik op deze manier opdoe ook in mijn werk meestal goed gebruiken.

"Hoe moet ik het schrijven?"

Ik denk dat deze vraag, samen met de twijfels over eigen kunnen het meeste voorkomt. Vaak zie je op een clubbijeenkomst iets leuks of je hoort dat iemand iets leuks met zijn computer doet. Als je dan vraagt of de persoon er iets over wil schrijven, dan zegt men "Ja, dat zal ik eens doen" waarna je er niets meer van hoort. In gesprekken blijkt dan meestal dat men niet weet hoe men moet beginnen of hoe je een artikel opbouwt. Welnu, dat is eigenlijk heel simpel. Je begint bij het begin (de inleiding) en je eindigt aan het eind (de afsluiting). Verder maak je voor je zelf een schema waarin je het onderwerp verdeelt in kleine onderwerpjes die dan later in je artikelje de kopjes van de paragrafen worden. Dus als voorbeeld:

Hoe schrijf je een artikel voor de μ P Kenner?

- + --- Inleiding.
- + --- "Kan ik het wel?"
- + --- "Waarover zal ik schrijven?"
- + --- "Hoe moet ik het schrijven?"
- + --- Hulp van de redactie
 - + --- Hulp bij de keuze van het onderwerp
 - + --- Hulp bij de benodigde literatuur
 - + --- Redigeren van het concept
 - + --- Layouten
- + --- Wijze van inleveren
- + --- Afsluiting

Voor de wat grotere artikelen kun je een paragraaf weer verdelen in nog kleinere stukjes die je desnoods weer opdeelt. Je maakt als het ware een inhoudsopgave vooraf van het artikel dat je gaat maken. Deze manier van werken kennen de programmeurs onder ons als de "Top down" methode; je splitst het probleem (of het artikel) op in steeds kleinere brokjes totdat je stukjes over houdt die je goed kunt overzien. Sommige tekstverwerkers hebben voor dit proces een hulpmiddel die "Outliner" genoemd wordt en waarmee je snel je ideeën in een schematische structuur kunt zetten. Uiteraard kun je dat ook op een stukje papier, met een potlood doen. Op deze manier kun je, als je eenmaal een idee hebt, desnoods een hele verzameling boeken gaan schrijven ("In de ban van de ring" of "de Aardkinderen").

Goed, nadat je grofweg de structuur van je artikel hebt bepaald, ga je de paragrafen vullen. Deel binnen een paragraaf je tekst op in alinea's zodanig dat een alinea in enkele zinnen iets vertelt. Binnen een alinea moeten de zinnen op een logische manier bij elkaar horen. Maak verder de zinnen niet te lang, gebruik komma's en vermijd moeilijke woorden.

Op welke manier je iets schrijft is meer een persoonlijke zaak. Ik doe meestal net of ik een brief aan iemand schrijf waarin ik het onderwerp probeer uit te leggen. Wat je wel in de gaten moet houden is dat er een duidelijk verschil is tussen spreektaal en schrijftaal. In een geschreven stuk is het belangrijk dat je vlot lopende zinnen gebruikt. Je kunt dat vaak het beste ontdekken als je jezelf het artikel hardop voorleest. Als er dan iets niet helemaal lekker loopt, merk je dat meteen zodat je het kunt corrigeren.

Uiteraard is het wenselijk een artikel op te fleuren met een aantal illustraties. Dit kunnen tekeningen zijn of foto's die door de auteur zelf gemaakt zijn, maar ook kun je figuren overnemen uit boeken en artikelen van anderen. Hier krijg je uiteraard meteen te maken met het copyright. Ik geloof dat het in het algemeen toegestaan is enkele figuren over te nemen mits er een duidelijke bronvermelding bij staat. Het is uiteraard nooit toegestaan werk van anderen als eigen werk te publiceren. In de praktijk verwacht ik niet dat er problemen zullen ontstaan als je bij een artikel in de μ P Kenner enkele figuren over het betreffende onderwerp uit boek XYZ overneemt.

Als je figuren opneemt zorg dan voor een pakkend onder- of bijschrift en verwijs in je artikel ook naar de figuren. Hetzelfde geldt ook voor listings en tabellen die ook als figuren beschouwd kunnen worden.

Goed, nadat het hele artikel ingetikt of opgeschreven is, lees je het nog enkele malen aandachtig door. Het beste kun je dat hardop doen omdat je dan meteen merkt welke zinnen niet lekker lopen. Als laatste stap haal je het artikel door de spellingchecker van de tekstverwerker waarna je het artikel opstuurt naar de redactie.

Hulp van de redactie

Uiteraard is de redactie (en daar reken ik mijzelf ook nog maar even bij) bereid hulp te bieden bij het schrijven van een artikel. In de eerste plaats kan dit al bij de keuze van het onderwerp. Vooral als u een uitgebreide artikelenreeks wilt gaan schrijven is het verstandig even met de redactie te overleggen. Verder kan het natuurlijk voorkomen dat de redactie iemand benadert met de vraag of hij eens iets over een bepaald onderwerp wil gaan schrijven.

Wat ook zou kunnen is dat de redactie benadert wordt met de vraag of zij aan bepaalde informatie kan komen. Ik zou mij voor kunnen stellen dat iemand een artikel wil schrijven over een IBM AS/400 maar geen plaatje van dergelijke machine heeft om in het artikel op te nemen. In dat geval kun je natuurlijk de redactie benaderen om te vragen of die misschien aan zo'n plaatje kan komen. Hetzelfde geldt voor benodigde technische informatie. De redactie is altijd bereid informatie die voor een artikel nodig is (bijvoorbeeld data sheets) te proberen te bemachtigen.

Een derde punt waar de redactie hulp kan bieden is het redigeren van het concept. Het komt wel eens voor dat iemand een concept artikel geschreven heeft waarvan de redactie zegt: "Het is een leuk onderwerp, alleen het artikel zou iets anders geschreven moeten worden". In dat geval kan de redactie, in overleg met de auteur, het artikel aanpassen of aangeven wat er veranderd zou moeten worden. Kleine veranderingen (punten, komma's d's en t's en kleine veranderingen de bouw van een zin) doet de redactie meestal op eigen initiatief, maar ik neem niet aan dat daar bezwaar tegen is.

Nadat een artikel voor plaatsing geaccepteerd is, wordt het opgemaakt. Dit betekent dat de layouter de tekst en de figuren krijgt en hiervan pagina's maakt die later samengevoegd worden tot een blad. Dit betekent ook dat de auteur zelf zich zo weinig mogelijk met de layout zou moeten bemoeien. De tekst kan het beste aan één stuk worden ingeleverd waarbij de paragrafen worden voorzien van een

kopje en de alinea's worden gescheiden door een lege regel. Gebruik verder geen opmaak-attributen zoals vet, onderstrepen en inspringen. Hebt u speciale wensen, geef dat dan duidelijk in de tekst aan zoals bijvoorbeeld "Layouter, de volgende kolommen s.v.p. netjes uitlijnen". Of "Layouter, hier zo ongeveer figuur 1 plaatsen, bijschrift: etc.". Maak dergelijke opdrachten voor de layouter duidelijk herkenbaar, bijvoorbeeld door ze vet en onderstreept in de tekst op te nemen.

Wijze van inleveren

Goed, je hebt een artikeltje geschreven en wilt het inleveren bij de redactie. Het beste is dan dat u het artikel in magnetische vorm aanbiedt. Dit mag in de vorm zijn van een tekstverwerkingsbestand (bij voorkeur WP 4.2) maar ook als zogenaamd ASCII bestand. In het eerste geval mag er alleen een zogenaamde harde return aan het eind van een alinea staan; indien u gebruik maakt van het ASCII formaat moeten de alinea's van elkaar gescheiden zijn door middel van twee returns!

Ook figuren kunnen het beste in magnetische vorm worden aangeleverd. Hierbij gaat de voorkeur uit naar lijntekeningen in het zogenaamde HPGL formaat. Hebt u de figuren niet in magnetische vorm maak dan fotokopieën met een goede zwart/wit verhouding en stuur die op. De layouter zorgt dan voor een kader van de juiste afmetingen waarna de drukker de figuren inplakt. Geef bij de figuren

duidelijk aan welk nummer het figuur in de tekst krijgt.

Omdat listings ook als figuur behandeld worden en ze worden afgedrukt in een ander lettertype, moeten deze ook als apart bestand aangeleverd worden. Wilt u stukjes listing in de tekst tussenvoegen geef dat dan met opdrachten voor de layouter aan.

Als je de spullen bij elkaar hebt, kun je ze uploaden naar The Ultimate of opsturen naar het redactiedres.

Afsluiting.

Ik hoop dat ik met dit stukje enkele mensen over de streep kan trekken om nu toch eindelijk dat artikel te gaan schrijven dat ze al zo lang wilden schrijven. Mocht je daarbij hulp nodig hebben of wil je op een andere manier op dit artikel reageren, dan nodig ik

**Layouter, de volgende
kolommen s.v.p. netjes
uitlijnen. Of:
Layouter, hier zo
ongeveer figuur 1
plaatsen.**

je dringend uit om contact met mij of de redacteur op te nemen.

Tenslotte wil ik iedere aspirant auteur nog een hart onder de riem steken: Wees niet bang fouten te maken. Ik lees liever een artikel waar de zinnen niet zo lekker lopen dan dat er over het betreffende on-

derwerp helemaal geen artikel in de μP Kenner terecht komt.

Veel succes!

Gert van Opbroek

Naschrift van de redactie

Indien u uw artikelen niet in WP 4.2 formaat aan kunt leveren of uw tekenpakket geen HPGL kan genereren, kunt u in overleg met de redactie uw teksten en tekeningen in één van de volgende formaten aanleveren:

Tekstfiles: WordStar 3, WordStar 4.0 of 5.0, XYWrite, MS-Word, Writer, MultiMate, DCA of WordPerfect 5.0

Lijntekeningen: GEM, AutoCad .SLD, MAC Picture, VideoShow, CGM, MS Windows, PostScript en tenslotte Lotus .PIC-files.

Rastertekeningen: GEM/HALO DPE, PC Paint-Brush (PCX), MAC Paint of TIFF.

Teksten en/of tekeningen die zonder aanduiding van het gebruikte formaat worden aangeleverd komen helaas niet in aanmerking voor plaatsing.

Bij lijntekeningen dient u er op te letten dat een aantal pakketten niet juist omgaan met "vreemde" formaten. Ook kan font-informatie over het algemeen niet ge-exporteerd worden naar een ander formaat. Heeft u tekeningen met fonts erin? Draai de tekening dan uit op 300 DPI (meer mag ook...) en stuur hem op!

Bij rastertekeningen tenslotte dient u er rekening mee te houden dat uw tekeningen niet zonder verlies aan kwaliteit (en soms ook aan beeldinformatie!) vergroot of verkleind kunnen worden.

Indien uw tekenpakket geen van de genoemde formaten ondersteunt kunt u, naast de papieren vorm, de tekening ook in PCL (de taal van de HP LaserJet en DeskJet printers) aanleveren. Gebruik hiervoor bijvoorbeeld opties als "printen naar file" of gebruik een printer-redirector (Prn2File...). De redactie garandeert niet dat alle PCL files omgezet kunnen worden naar een rasterformaat, maar de meest gangbare commando's worden ondersteund.

Joost Voorhaar

(Advertentie)

Te Koop : DOS65 Computer + toebehoren

Bestaande uit :

- EPS 80024 - Bus kaart
- EPS 80089 - Junior Hoofd print
- EPS 81033 - Junior Interface print
- EPS 82010A - EPROMmer
- EPS 82017 - Dynamische RAM-kaart
1x 16kB, 1x 64kB
- EPS 82093 - mini-EPROM-kaart
- EPS 83014 - Universele geheugen kaart
incl. 24kB SRAM
- EPS 83082 - VDU kaart
- EPS 83101 - BASICODE-2 kaart
- EPS 83108 2 - VIA-ACIA kaart
- DOS65 floppydisk interface kaart
- RCA VP601 Toetsenbord
- ASTEC Geschakelde voeding
- Geluid interface kaart (SID-6581)

Ser/par omzetter t.b.v. IBM toetsenbord
ALTAI Computer Cassette Recorder
19" rek
Incl. alle (bouw)beschrijvingen, Junior Computer
boeken deel 1 t/m 4, VIA 6522 boek, DOS-65 software en documentatie

Prijs voor dit bovenstaand geheel : t.e.a.b.
Inlichtingen : Frank Bens

Tjalkstraat 25
1784 RX Den Helder
Tel: 02230-33379

Voor de liefhebber !!!!!

Oude clubbladen "de 6502 kenner", No. 20 t/m 58
t.w. febr '82 t/m okt '88
Adres: zie boven.

Trace exception routine voor MC68000

Bij 680x0 machines krijgt men wel eens "bommeldingen". Deze waarschuwingen laten weten dat er iets mis is. De "bus error" komt bijvoorbeeld voor als het gevolg van een pointer die buiten het werkgeheugen wijst. De machine zal niet zomaar gaan "hangen", maar wat er precies fout ging weet je ook weer niet. Terwijl de processor wel de mogelijkheid biedt de oorzaak nader te achterhalen; er worden diverse (belangrijke) gegevens op de stack gezet (zie figuur 1). Ook bij een zogenaamde "address error" worden deze gegevens op de stack beschikbaar gemaakt.

Spoor volgen

Met die twee exception routines en een "putchar" op geschikte plaatsen kun je een programma min of meer debuggen. De processor zelf biedt nog een andere mogelijkheid, het zogenaamde "Tracing". In de trace mode wordt na iedere instructie een exception gegenereerd. Het is dan mogelijk de inhoud van bijvoorbeeld de verschillende registers af te drukken. De trace mode gebruikt het T-bit in het supervisor deel van het statusregister. Als het T-bit op nul staat, is de tracing uitgeschakeld en worden de instructies gewoon na elkaar uitgevoerd. Als het T-bit geset is aan het begin van een instructie, zal een trace exception gegenereerd worden na het beëindigen van de complete instructie.

Trace exception processing

Het verloop van een trace exception gaat als volgt. Na de executie van de instructie en vóór de start van de volgende, wordt een kopie gemaakt van het statusregister. Er wordt overgeschakeld naar supervisor mode en de trace mode wordt uitgeschakeld om geneste tracing te voorkomen. Het trace vector nummer wordt gegenereerd, en program counter en de kopie van het status register worden op de supervisor stack gepusht. De processor gaat vervolgens de

instructie uitvoeren die door de trace vector aangewezen wordt.

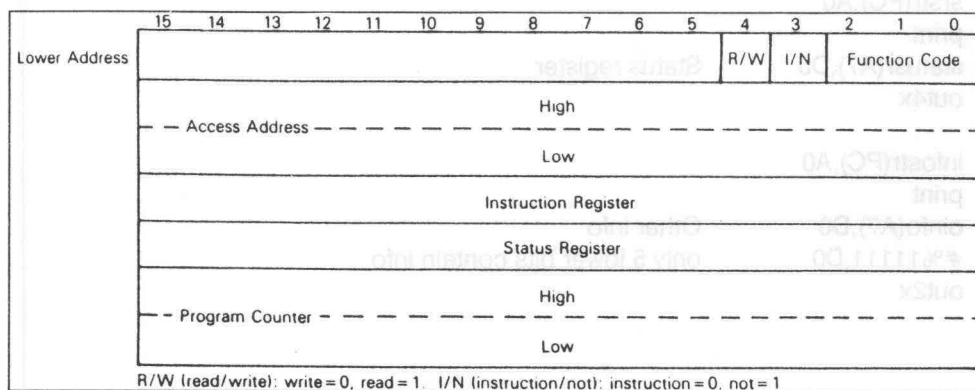
Source listing

In figuur 2 staat een source listing van de bedoelde exception routines. Deze routines kunnen door de KGN68k assembler omgezet worden naar motorola S-records die op hun beurt gedownload kunnen worden naar de target machine. Het statement "use ASCII.DEF" zorgt er voor dat de file ASCII.DEF gelezen wordt en dat constanten zoals "CR" en "LF" gedefinieerd worden. Dan volgen er wat systeemdefinities (zie figuur 1). De vaste plaatsen in

RAM zijn alleen nodig voor de opslaan van registers, en de stack zit natuurlijk in RAM. Het verschil in vectors van bus en address errors wordt slechts gebruikt om verschillende teksten af te drukken; voor de rest zijn ze aan elkaar gelijk. De trace exception bestaat eigenlijk alleen uit een flinke reeks putchars; de processor zorgt voor de rest. De volgende twee routines zijn nodig om de trace mode in of

uit te schakelen. De correcties van de program counter, die op de stack staat, zijn nodig omdat line A en line F emulators "instructies" kunnen bevatten die langer zijn dan 1 word. Bij deze manier moet de programmeur altijd het terug adres aanpassen. Hij hoeft dus niet te "smokkelen" als het om een langere "instructie" gaat. De uitvoerroutines en de text strings zullen geen uitleg nodig hebben. Als laatste het belangrijkste: de vectoren. Met behulp van deze vectoren wordt de processor verteld waar de betreffende routine staat.

**Bij 680x0 machines
krijgt men wel eens
"bommeldingen".**



**Fig. 1: Supervisor
Stack Order for Bus
or Address Error**

*

* Some useful Exception routines

*

use ASCII.DEF

*

EQUATIONS

RAM

EQU

\$78000

ROM

EQU

\$70000

monitor

EQU

\$FC0870

warm start

oinfo

EQU

0

other informatie

accesa

EQU

2

Accessed Address

instrct

EQU

6

executed Instruction when error occurred

statusr

EQU

8

status register

progcnt

EQU

10

program counter

regs

org

RAM

ds.l

16

for all Data & Address registers

rorg

ROM

bus_err

movem.l

d0-d7/a0-a7,regs

saves the registers

lea

bus_text(PC),A0

bsr

print

bra

baeh

Do bus & adres error handling

*

adr_err

movem.l

d0-d7/a0-a7,regs

saves the registers

lea

adr_text(PC),A0

bsr

print

*

* Bus & Adres error handling

baeh

move.l

accesa(A7),D0

Accessed Address from stack

bsr

out8x

lea

pcstr(PC),A0

bsr

print

move.l

progcnt(A7),D0

Programm counter

bsr

out8x

lea

instr(PC),A0

bsr

print

move.w

instrct(A7),D0

Instruction

bsr

out4x

lea

srstr(PC),A0

bsr

print

move.w

statusr(A7),D0

Status register

bsr

out4x

lea

infostr(PC),A0

bsr

print

move.w

oinfo(A7),D0

Other info

andi.w

##%11111,D0

only 5 lower bits contain info

bsr

out2x

```

        bsr      p16regs

        adda.l   #14,A7          clean up the stack
        jmp      monitor        warm start

*****
*
*   TRACE Exception
*
*****
xtrace    movem.l d0-d7/a0-a7,regs
        bsr      p16regs

; Print PC, status register, and USP
        lea      srstr(PC),A0
        bsr      print
        move.w   (A7),D0        DO = Status Register
        bsr      out4x
        lea      pcstr(PC),A0
        bsr      print
        move.l   2(A7),D0       Program Counter
        bsr      out8x
        lea      instr(PC),A0
        bsr      print
        move.l   2(A7),A0       Program Counter
        move.w   (A0),D0       (First word of) Instruction
        bsr      out4x
        lea      uspstr(PC),A0
        bsr      print
        move.l   USP,A0
        move.l   A0,D0
        bsr      out8x
        bsr      crlf
        movem.l  regs,D0-D7/A0-A7
        rte

*****
*
*   LINE A Exception
*
*****
; turns Trace mode OFF
line_a    move.l   D0,-(A7)      save D0
        move.w   4(A7),D0       get Status Register
        andi.w   #$7FFF,D0      clear Trace Bit
        move.w   D0,4(A7)       putback SR
        move.l   6(A7),D0       get Programm counter
        add.l    #2,D0           increment Programm counter
        move.l   D0,6(A7)       putback P.C.
        move.l   (A7) + ,D0      restore D0
        rte

```



```

*****
*
* LINE F Exception
*
*****

; turns Trace mode ON (enable)
line_f      move.l    D0,-(A7)      save D0
            move.w    4(A7),D0      get Status Register
            ori.w     #$8000,D0     set Trace Bit
            move.w    D0,4(A7)      putback SR
            move.l    6(A7),D0      get Programm counter
            add.l     #2,D0          increment Programm counter
            move.l    D0,6(A7)      putback P.C.
            move.l    (A7)+,D0      restore D0
            rte

p16regs     bsr       crlf           "D "
            lea       dstr(PC),A0
            bsr       print
            lea       regs,A1        A1 - DATA registers
            bsr       p8regs         print DATA registers
            lea       astr(PC),A0    "A "
            bsr       print
            lea       regs+32,A1     ADDRESS registers
            bsr       p8regs         print them
            rts

p8regs      move.w    #7,D1
loop        move.l    (A1)+,D0
            bsr       out8x
            move.b    #' ',D0
            bsr       Putchar
            dbra      d1,loop
            bsr       crlf
            rts

Putchar     jmp       $FC072C
out2x       jmp       $FC07B8        prints a byte hexadecimal
out4x       jmp       $FC07C0        p. a word hex
out8x       jmp       $FC07C8        p. a long hex
print       jmp       $FC07DA        prints a string
ps          move.l    D0,-(A7)
            move.b    #' ',D0
            bsr.s     Putchar
            move.l    (A7)+,D0
            rts

crlf        lea       crlf_text(PC),A0
            jmp       print

*
* Text strings
*
bus_text    dc.b      CR,LF,"bus error at :",EOT
adr_text    dc.b      CR,LF,"Address error at :",EOT

```

```

crlf_text    dc.b    CR,LF,EOT
dstr         dc.b    "D",EOT
astr         dc.b    "A",EOT
srstr        dc.b    "SR=",EOT
pcstr        dc.b    "PC:",EOT
uspstr       dc.b    "USP=",EOT
instr        dc.b    "instr=",EOT
infostr      dc.b    "info:",EOT

```

*

* Vectors

*

```

RORG    8
DC.L    bus_err
DC.L    adr_err
RORG    $24
DC.L    xtrace
DC.L    line_a
DC.L    line_f
END

```

* End of file

Praktijk

Het programmaatje uit figuur 3 geeft dan het resultaat dat in figuur 4 staat. Daar waar "dc.w \$123" staat wordt het trace bit veranderd. De waarde \$123 is willekeurig, het is een unieke tekencombinatie die gemakkelijk met een zoekfunctie van een editor is op te sporen. De trace exception wordt dus uitge-

voerd zolang het T-bit "1" is. Nadat er een oneven getal (D3=3) bij A4 was opgeteld, ging move D3,(A4) pas fout. Terug bij de monitor prompt kan men dan andere debugstappen ondernemen.

Geert Stappers

```

1  fffffff  org    $20000
2  00020000 163C0003 start  move.b #3,D3
3  00020004 227C00001111 move.l $1111,A1
4  0002000a F123      dc.w    $f123
5  0002000c 287C00004444 move.l #$4444,A4
6  00020012 D3CC      add.l   A4,A1
7  00020014 2E09      move.l   A1,D7
8  00020016 E38F      lsl.l    #1,D7
9  00020018 A123      dc.w    $a123
10 0002001a E7BF      rol.l    D3,D7
11 0002001c 3883      move.w   D3,(A4)  eerste keer
12 0002001e D9C3      add.l    D3,A4
13 00020020 3883      move.w   D3,(A4)  tweede keer
14 00020022 4E71      nop
15 00020024 4E75      rts
16 00020026      end

```

Symbol table : start \$00020000

Fig. 3: ATEST.LST

```

g 020000
D 00000000 00020000 00000000 00000003 00000000 00000000 00000000 00000000
A 00FC09BC 00001111 00000000 00000000 00004444 00000000 00000000 000FFFDA
SR = A700 PC:00020012 instr = D3CC USP = FFFFFFFF

D 00000000 00020000 00000000 00000003 00000000 00000000 00000000 00000000
A 00FC09BC 00005555 00000000 00000000 00004444 00000000 00000000 000FFFDA
SR = A700 PC:00020014 instr = 2E09 USP = FFFFFFFF

D 00000000 00020000 00000000 00000003 00000000 00000000 00000000 00005555
A 00FC09BC 00005555 00000000 00000000 00004444 00000000 00000000 000FFFDA
SR = A700 PC:00020016 instr = E38F USP = FFFFFFFF

D 00000000 00020000 00000000 00000003 00000000 00000000 00000000 0000AAAA
A 00FC09BC 00005555 00000000 00000000 00004444 00000000 00000000 000FFFDA
SR = A700 PC:00020018 instr = A123 USP = FFFFFFFF

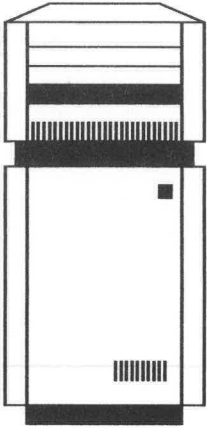


Address error at :00004447 PC:00020024 instr = 3883 SR = 2700 info:05
D 00000000 00020000 00000000 00000003 00000000 00000000 00000000 00055550
A 00FC09BC 00005555 00000000 00000000 00004447 00000000 00000000 000FFFD2

```

S-BUG 2.3 - 68000

Fig. 4: ATEST.LOG

BBS "The Ultimate" For All Systems

Telefoon 053-303902 & 053-328506

- 053-303902 - (2 lijnen!)

V22, V22bis, V23, V32bis, 14k4/HST, V42bis, MNP5

- 053-328506 -

V21, V22, V22bis, V23, V32bis, 14k4/HST, V42bis, MNP5

De Motorola 68030; het hart van KGN68k

Inleiding

In deze artikelenreeks wil ik enige aspecten gaan behandelen van de 68030 die, zoals bekend gekozen is als het hart van het nieuwe KGN-systeem. Omdat ik als zogenaamde "softwerker" meer verstand heb van de zachte kant van de processor, zal ik een uitgebreide beschrijving van de elektronische aspecten aan anderen overlaten.

Wat kun je zoal verwachten in deze serie? Uiteraard een uitgebreide beschrijving van het zogenaamde programmeermodel. Dit zal het hoofd-bestanddeel van dit deel van de serie zijn. Verder onder andere beschrijvingen van caching, de interruptstructuur en het gebruik van de Memory Management Unit. Tenslotte zal de serie naar verwachting parallel gaan lopen aan de ontwikkeling van de software voor KGN68k zodat daar ook nog wel enkele onderwerpen uit naar voren zullen komen.

De 68000-familie

De 68030 is een lid van een roemrijke familie van Motorola processoren. Zoals Intel een hele familie 80xxx processoren heeft, heeft Motorola een hele serie 68000 processoren. Hieronder volgt een kort overzicht van de diverse leden.

68008:

De kleinste telg uit het roemrijke geslacht is de 68008. Deze processor heeft de beschikking over een 8 bits databus en een 24 bits adresbus. Intern is de processor echter opgebouwd als een echte 32-bitter, de registers hebben een breedte van 32 bits.

68000:

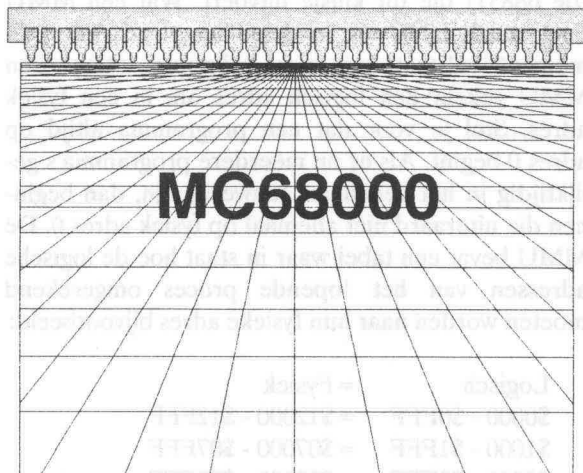
Dit is de stamvader van de familie. De processor heeft een 24 bits adresbus en een 16 bits databus. De 68000 loopt al weer zo'n tien jaar mee maar is nog steeds niet verouderd. Dit is de processor die te vinden is in de beroemde machines van Apple, Atari en Commodore.

68010:

Ten opzichte van de 68000 is de belangrijkste uitbreiding dat de processor enkele hulpmiddelen heeft waardoor hij om kan gaan met virtuele adressen. Verder is de processor bij gelijke kloksnelheid ongeveer 30% sneller dan de 68000.

68020:

Een hele belangrijke processor in de familie is de 68020. Deze processor komen we heel veel tegen in de wat duurdere 68k-machines die enkele jaren geleden ontworpen zijn. De processor heeft zowel een 32 bits brede adresbus als een 32 bits brede databus. Verder heeft de processor d.m.v. een zogenaamde Dynamic Bus Sizing ook de mogelijkheid



geheugengebieden met een breedte van 16 of 8 bits aan te spreken. Om goed samen te kunnen werken met coprocessors (bijvoorbeeld de 68882 mathematische coprocessor) heeft de processor hiervoor een speciaal interface. De 68020 heeft ten slotte een instructiecache op de chip.

68030:

De 68030 is een processor die langzaam maar zeker ook in de kant en klare systemen op de markt komt. Deze processor is weer een uitbreiding op de 68020. De 68030 is een 68020 met behalve een instructie cache, ook nog een data cache en een geïntegreerde Memory Management Unit. De 68030 is de processor die ingezet wordt voor KGN68k.

68040:

De jongste en meest krachtige telg uit de familie is de 68040. Behalve alle eigenschappen van de 68030 en de 68020 is de 68040 ook nog een rekenwonder. De processor heeft namelijk een geïntegreerde Floating Point Unit. Voor KGN68k was deze chip ons echter net iets te zwaar (en te duur).

68881/68882:

Alleen het rekenwonder uit de familie de 68040 kan zelf allerlei gecompliceerde berekeningen uitvoeren. Voor de iets minder getalenteerde familieleden heeft Motorola een tweetal zakrekenmachines op de markt gebracht. Dit zijn de 68881 en 68882 mathematische coprocessors. De 68882 is volledig compatibel met de 68881, hij kost alleen wat meer en is een stukje sneller. Zoals bij de 68020 al opgemerkt is, hebben de 68020 en hoger een aparte coprocessor interface waardoor de coprocessor als gelijkwaardige partner van de processor functioneert. Bij de lagere processoren kan de coprocessor het beste als een periferie chip beschouwd worden.

68851:

De 68030 heeft een interne Memory Management Unit of MMU. Voor de 68020 is er een coprocessor (de 68851) die dit klusje uitvoert. Wat een MMU doet wordt later nog in detail uitgelegd. Voor dit moment volstaat een zeer beknopte uitleg. Een MMU rekent een logisch adres om in een fysiek adres. Stel je voor dat een programma altijd op adres 0 begint. Als er nu meerdere programma's gelijktijdig in het geheugen aanwezig zijn, dan beginnen die uiteraard niet allemaal op fysiek adres 0. De MMU bevat een tabel waar in staat hoe de logische adressen van het lopende proces omgerekend moeten worden naar hun fysieke adres bijvoorbeeld:

Logisch	= Fysiek
\$0000 - \$0FFF	=\$12000 - \$12FFF
\$1000 - \$1FFF	=\$07000 - \$07FFF
\$2000 - \$2FFF	=\$23000 - \$23FFF

Op deze manier kun je verschillende programma's gelijktijdig in het geheugen hebben die toch allemaal dezelfde logische adressen gebruiken. Bij het opstarten van een ander programma hoeft je alleen maar de tabel in de MMU te veranderen en je werkt met een ander programma. Dit is vooral erg handig bij zogenaamde Multi Tasking waarbij de processor

zijn tijd verdeelt over een aantal programma's. Ik kom daar in een later stadium nog uitgebreid op terug.

68230, 68681 etc.

Zoals elke adellijke familie heeft de 68000 familie ook wat huis-, tuin- en keukenpersoneel. Dit zijn bijvoorbeeld een Parallel Interface/Timer en een Dual Asynchronous Receiver/Transmitter die ook beide ingezet worden in KGN68k. Verder kan de familie zeer goed samenwerken met de periferie chips van andere dan Motorola fabrikanten zoals ook blijkt uit het ontwerp van KGN68k.

Varianten:

Behalve de bovengenoemde onderdelen zijn er nog een aantal varianten op deze chips. Dit zijn in de eerste plaats een aantal van de 68000-lijn afgeleide processoren die door Motorola zelf ontwikkeld zijn en in de tweede plaats vergelijkbare processoren van andere fabrikanten, zoals bijvoorbeeld de 68070 van Philips die veel lijkt op een uitgebreide 68000 en die onder andere in CD-spelers toegepast wordt.

Kenmerken van de 68030

Hieronder worden de belangrijkste eigenschappen van de 68030 opgesomd (bron: ref.1).

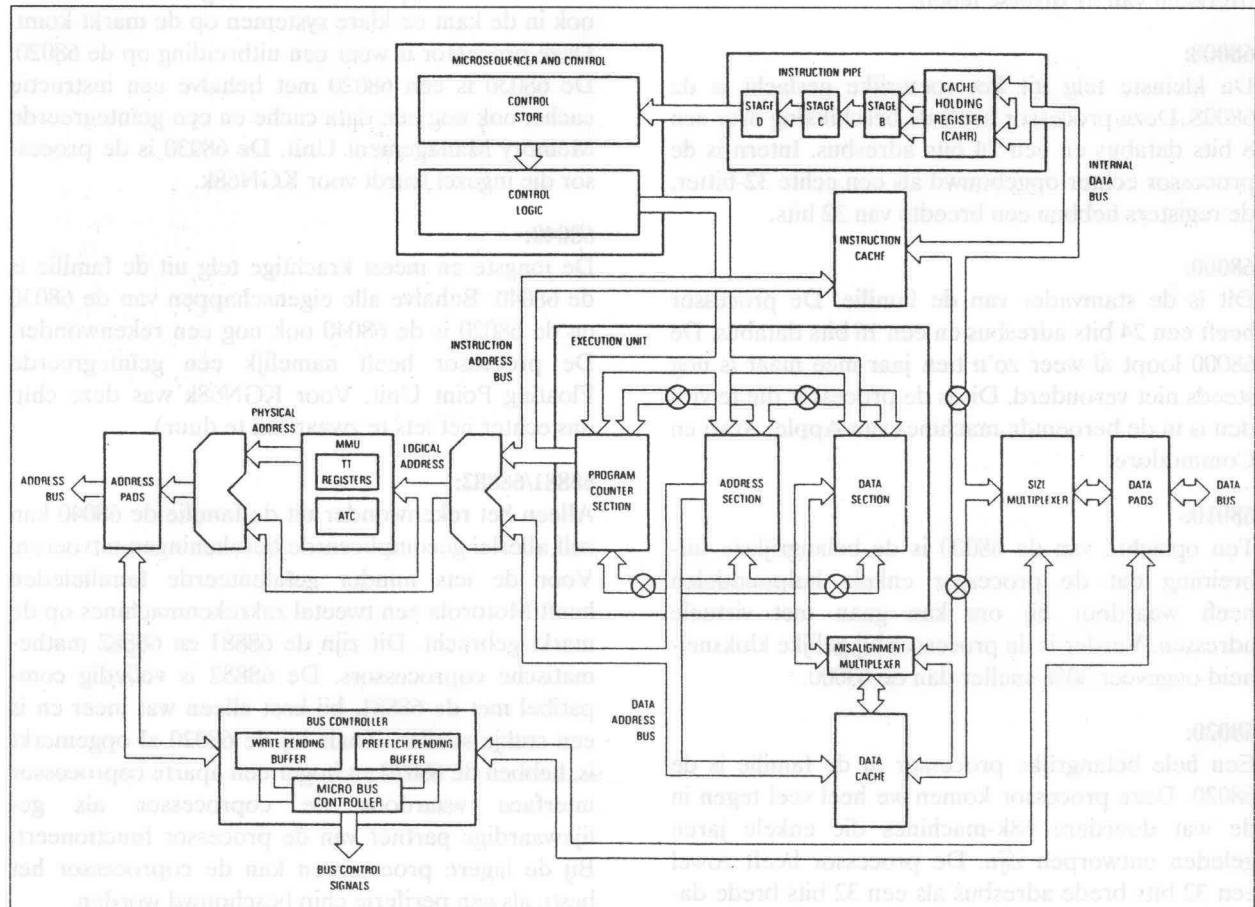


Fig. 1: interne opbouw van de 68030

- Object code compatibel met de 68020 en oudere versies uit de 68000 familie.
- Volledige 32 bits brede niet gemultiplexte adres- en databus.
- 16 32 bit brede General Purpose (algemeen toepasbare) registers.
- 256 byte instructie cache en 256 byte data cache die gelijktijdig benaderd kunnen worden.
- Een Paged MMU voor de transformatie van adressen, parallel aan de uitvoering van instructies en de benadering van de caches.
- Twee zogenaamde "Transparante secties" voor de rechtstreekse benadering van twee geheugengebieden om grote hoeveelheden data uit te wisselen voor bijvoorbeeld grafische toepassingen.
- Pipelined Architecture (Sorry, ik denk dat hiervoor geen goede vertaling bestaat) waardoor de processor aan meerdere instructies tegelijk kan werken.
- Een geavanceerde bus controller waarmee asynchrone (minimaal 3 clock cycles), synchrone (minimaal 2 bus cycles) en burst (minimaal 1 bus cycle) transfers mogelijk zijn over het hele fysieke adresbereik.
- Dynamic Bus Sizing waardoor ook periferie- en geheugenbouwstenen met een breedte van 16 of 8 bits benaderd kunnen worden.
- Ondersteunt coprocessors met het Motorola 68000 interface.
- Ondersteunt een logische en fysieke adresruimte van 4 Giga Byte (4 miljard byte).
- Is uitgevoerd in een combinatie van HMOS en CMOS.
- Is leverbaar voor de clock snelheden 20 MHz, 25 MHz en 33 MHz.

Interne opbouw

Uiteraard zijn er binnen de 68000-familie een groot aantal gemeenschappelijke kenmerken. Er is zeggend een sterke familieband. Dit gaat zover dat de processoren upwards compatibel met elkaar zijn hetgeen wil zeggen dat code voor een 68008 draait op en 68000 en zelfs op een 68030 of 68040. Uiteraard gaat het dan allemaal wel wat sneller. Kijken we naar het aantal instructies dat de 68030 heeft in vergelijking met de 68000, dan verschilt dat maar enkele instructies. De 68020 en 68030 hebben wel wat meer mogelijkheden bij de adressering. Verder heeft een 030 een aantal hulpmiddelen, in de vorm van caches en MMU, aan boord om de zaken gemakkelijker en/of sneller te maken.

In figuur 1 is de interne opbouw getekend van de 68030. Aan de hand van dit diagram maken we in dit artikel een korte excursie door de processor. In de volgende afleveringen zullen de diverse onderwerpen dan wat verder uitgediept worden.

De execution unit

De execution unit is het eigenlijke hart van de processor. Deze eenheid bevat de program counter of PC die aangeeft welke instructie uitgevoerd gaat worden, een eenheid om adressen te berekenen en een eenheid waar de data intern in de processor opgeslagen kan worden. Deze laatste eenheid bevat de meeste zogenaamde registers waarvan de 68030 er een groot aantal in diverse smaken heeft.

Alle processoren uit de 68000-familie bezitten 8 adres- (A0 t/m A7) en 8 dataregisters (D0 t/m D7). Deze registers hebben een breedte van 32 bits waarbij een dataregister ook gebruikt kan worden voor 1 byte, 2 byte (word) en 4 byte (long word) operaties. Een adresregister kan gebruikt worden voor word en long word operaties. Een dataregister kan met behulp van speciale instructies ook gebruikt worden voor zogenaamde bit fields van 1 t/m 32 bits.

In figuur 1 zijn de adres- en dataregisters uiteraard te vinden in de blokjes "Address Section" en "Data Section". Het blokje "Misalignment Multiplexer" heeft als taak het mogelijk te maken een byte of word vanaf een andere plaats dan vanaf het begin van een long word te benaderen.

In de praktijk behoort men een dataregister te gebruiken voor (hoe kan het ook anders) data. Een adresregister behoort men te gebruiken voor het opslaan van adressen. Dit kan bijvoorbeeld het startpunt in het logische geheugen zijn van de één of andere datastructuur. Een bijzonder geval is adresregister A7. Deze bevat een verwijzing naar de zogenaamde system stack. De stack loopt bij een 68000 van hoog naar laag geheugen waarbij A7 het adres bevat van het laatste brokje informatie dat op de stack geschreven is. Vanwege de verwijzing naar de system stack wordt A7 ook wel de Stack Pointer genoemd. Veel assemblers kennen daarom ook de alias SP wanneer A7 bedoeld wordt.

Voor een simpel programma lijkt het alsof een 68030 slechts één system stack heeft. Voor dat simpele programma is dat ook genoeg. De 68000 heeft er echter twee en de 68030 zelfs drie. Die ene stack die het simpele programma ziet is de zogenaamde User Stack. Het top-adres van deze stack staat in A7 als de processor in de zogenaamde User State is. A7 bevat dan de zogenaamde User Stack Pointer of USP. In deze toestand mogen ook niet alle beschikbare instructies gebruikt worden. Na een exception (een onderbreking van het lopende programma d.m.v. een externe gebeurtenis) komt de processor in de zogenaamde Supervisor State. In deze toestand staat in A7 het adres van één van de twee andere stacks. Afhankelijk van een bit in het status register staat de Interrupt Stack Pointer (ISP) in A7 of de

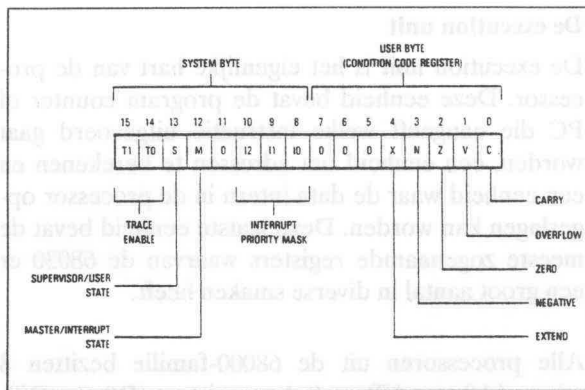


Fig. 2: statusregister van de 68030

Master Stack Pointer of MSP. In de Supervisor State mag de processor alle beschikbare instructies uitvoeren.

De USP en de User State zijn uiteraard bedoeld voor de normale programma's. De Supervisor State is bedoeld voor zaken die binnen het operating system liggen zoals het afhandelen van interrupts en het activeren van user programma's. Hierbij is de ISP bedoeld om als stack te dienen bij het afhandelen van interrupts. De MSP kan gebruikt worden om informatie over de programma's (taken) die in het geheugen aanwezig zijn snel te kunnen benaderen. De 68000 kent geen MSP zodat daar de verdeling eenvoudig zo ligt: De USP voor de gebruikers-programma's, de ISP voor het operating system.

Non Maskasble Interrupt, afgekort NMI (spreek uit: enemy = vijand)

In figuur 2 is het zogenaamde status woord van de 68030 getekend. Dit register is 16 bits breed (één woord) en is opgedeeld in een zogenaamd User Byte en een System Byte. Het User Byte wordt ook wel het Condition Code Register (CCR) genoemd. In dit CCR staan de zogenaamde vlaggen die aangeven wat het resultaat van de laatste instructie was. Voor assembler-programmeurs zijn deze vlaggen bijna allemaal gesneden koek, voor de andere lezers volgt toch nog even een overzicht:

De Carry vlag, C, geeft aan of er bij de laatste instructie een zogenaamde carry opgetreden is. Als voorbeeld:

```
move.l    #220,d0; vul d0 met het getal 220
addi.b    #100,d0; tel er 100 bij
```

Omdat bij de laatste (byte) operatie de grens van 255 (alle bits in het byte geset) overschreden is,

wordt de carry vlag gezet. Hetzelfde treedt op als we van het getal 100 het getal 120 aftrekken.

De Overflow vlag, V, geeft een zogenaamde overflow conditie aan. Een overflow conditie krijgen we bijvoorbeeld als we in een byte de getallen 100 en 110 bij elkaar optellen. Als resultaat krijgen we dan namelijk het getal 210 terwijl het grootste signed getal 127 is. Dit levert dan een overflow conditie op.

De Zero vlag Z: Het resultaat van de laatste instructie was nul.

De Negatief vlag N: Het resultaat van de laatste instructie was negatief.

De Extend of X vlag. Bij de normale instructies wordt de carry niet meegenomen bij de bewerking. Zo telt de ADD-instructie de twee operanden bij elkaar zonder acht te slaan op de carry. Hetzelfde geldt voor de rotate-instructies. Om toch met een carry te kunnen werken, heeft de 68000-familie een extra vlag die bij veel instructies een kopie van de C vlag bevat. Verder heeft de 68000-familie speciale instructies voor bewerkingen met deze zogenaamde Extend (X) vlag. Bijvoorbeeld ADDX, en ROXL.

Het system byte bevat nog zeven vlaggen. Deze zijn bij de 68030 alleen te benaderen als de processor in de Supervisor State staat. De instructies om het system byte te benaderen zijn namelijk geprivilegieerd hetgeen betekent dat ze niet in de User State uitgevoerd mogen worden. Probeert de programmeur dat toch, dan voert de processor een zogenaamde exception uit waarbij het lopende programma wordt onderbroken en het operating system wordt aangeroepen om bijvoorbeeld een foutmelding te geven. Ik zal op deze materie nog uitgebreid terugkomen.

De laagste van de zeven vlaggen geven de zogenaamde prioriteit aan waarin de processor zich bevindt. De 68030 kent namelijk 7 niveaus van interrupt, genummerd van 1 t/m 7. Hierbij kun je interrupts het beste voorstellen als gebeurtenissen die je werk kunnen onderbreken. In een computer zijn dit gebeurtenissen die van een randapparaat, bijvoorbeeld een I/O chip komen. Aan deze gebeurtenissen kun je een prioriteit toekennen. Als voorbeeld kun je mij op dit moment nemen. Ik ben een artikel aan het schrijven. Als nu mijn vrouw binnenkomt met een kopje koffie, dan krijgt dat prioriteit 5. Het begin

van een voetbalwedstrijd op televisie krijgt prioriteit 2 en de telefoon die gaat heeft vrijwel altijd prioriteit 7. Nu kun je de 68030 bijvoorbeeld instellen op prioriteit 3. Hij zal dan wel reageren op de koffie en de telefoon, maar niet op de voetbalwedstrijd. De 68030 laat zich als het ware alleen onderbreken door interrupts die een hogere prioriteit hebben dan die waar hij zelf in staat. Interrupt 7 is een speciale, die onderbreekt de processor altijd, ongeacht de instelling van de processor en wordt daarom ook wel Non Maskable Interrupt (NMI spreek uit: enemy = vijand) genoemd.

De M-vlag geeft aan of de processor, in Supervisor State, gebruik moet maken van de Interrupt Stack Pointer of de Master Stack Pointer.

De S-vlag geeft aan of de processor in Supervisor State is. Om deze vlag om te kunnen zetten, heb je een Privileged instructie nodig. Je kunt de Supervisor State dus wel uitzetten, maar niet zelf aanzetten. Na een exception (bijvoorbeeld een interrupt), staat de processor echter automatisch in Supervisor State.

Tenslotte heeft de processor nog twee trace vlaggen (T0 en T1). Dit zijn vlaggen met behulp waarmee een (assembler-) programmeur de fouten uit zijn programma kan halen. Als vlag T1 geset wordt (en T0 blijft clear), dan zal de processor na het uitvoeren van een instructie een exception uitvoeren waarbij het programma onderbroken wordt en er een bepaalde routine uit het operating system aangeroepen wordt. Deze routine kan dan bijvoorbeeld laten zien welke instructie uitgevoerd is en wat de inhoud van de diverse registers geworden is. Wordt alleen T0 geset, dan wordt de exception alleen uitgevoerd bij die instructies waarbij naar een ander deel van het programma gesprongen wordt. De combinatie van beide vlaggen geset wordt bij de 68030 niet gebruikt.

Het belangrijkste register voor de uitvoering van een programma hebben we nog niet gehad. Het betreft de Program Counter of PC. In dit register houdt de processor bij wat het adres is van de volgende instructie.

Met de bespreking van de adres- en dataregisters, het status register en de program counter hebben we het hele zogenaamde User Programmeermodel (zie figuur 3) gehad. Dit is het beeld dat de processor aan de gebruiker (het programma) in User State toont. We hebben zelfs al een uitstapje gemaakt naar het Supervisor Programmeermodel met de bespreking van de Interrupt Stack Pointer, de Master Stack Pointer en het system byte in het status register.

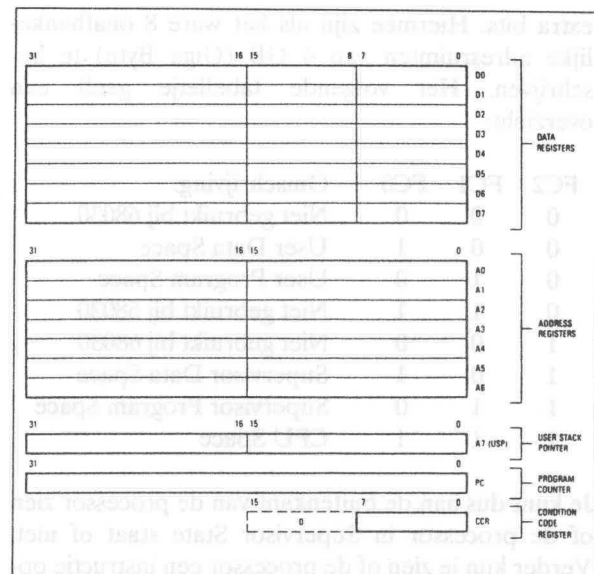


Fig. 3: user programmeermodel

Het Supervisor Programmeermodel heeft, naast alle registers van het user model een groot aantal extra registers. Veel van die registers hebben te maken met andere zaken dan de execution unit en zullen daar behandeld worden. In deze paragraaf worden er nog een paar behandeld.

Het eerste register dat nog behandeld moet worden is het Vector Base register. In de bespreking is al een aantal malen het woord "exception" gevallen. Dit is een gebeurtenis waarbij de processor het lopende programma onderbreekt en iets anders gaat doen, d.w.z. een stuk programma gaat uitvoeren dat op een ander adres in het geheugen staat. Nu moet de processor uiteraard wel weten waar dat stuk programma in het geheugen staat. Welnu, deze adressen staan in een tabel ergens in het geheugen van de processor. Verder zijn alle gebeurtenissen genummerd zodat de processor bij een bepaalde gebeurtenis weet op welke plaats hij in de tabel moet kijken. Dit nummer wordt de "vector" van de gebeurtenis (exception) genoemd. Het gaat zelfs zo ver dat bij interrupts het onderdeel dat aandacht vraagt (de telefoon) zelf aan kan geven wat zijn vector is. Bij de 68000 staat de tabel met adressen, de vector-tabel, gewoon op adres 0. Vanaf de 68010 heeft de processor in zijn programmeermodel een zogenaamd Vector Base Register waarin het adres geschreven kan worden van het startpunt van de vector tabel.

Wat ook nog bij de execution unit besproken moet worden zijn de Alternatieve Functie Code registers SFC en DFC. De functie codes zijn 3 signalen die aan de buitenkant van de processor op drie zogenaamde Functie Code pennen aanwezig zijn. Strikt genomen kunnen deze 3 signalen beschouwd worden als een uitbreiding van de adresbus met drie

extra bits. Hiermee zijn als het ware 8 onafhankelijke adresruimten van 4 GB (Giga Byte) te beschrijven. Het volgende tabelletje geeft een overzicht:

FC2	FC1	FC0	Omschrijving:
0	0	0	Niet gebruikt bij 68030
0	0	1	User Data Space
0	1	0	User Program Space
0	1	1	Niet gebruikt bij 68030
1	0	0	Niet gebruikt bij 68030
1	0	1	Supervisor Data Space
1	1	0	Supervisor Program Space
1	1	1	CPU Space

Je kunt dus aan de buitenkant van de processor zien of de processor in Supervisor State staat of niet. Verder kun je zien of de processor een instructie ophaalt of gegevens met het geheugen uitwisselt. Op deze manier zou je aparte geheugens voor programma, data in User State en in Supervisor State kunnen gebruiken. De laatste, de CPU space geeft aan dat de processor niet met het geheugen communiceert, maar met een co-processor. Ook op deze materie kom ik in een latere aflevering nog terug.

Goed, we weten nu wat de functie codes zijn, maar daarmee zijn de SFC en DFC registers nog steeds niet besproken. Welnu, er zijn enkele instructies waarbij de functie codes niet door de processor bepaald worden maar waarbij je ze op kunt geven. In dat geval staan de te gebruiken functie codes in het zogenaamde Source Function Code register voor de eerste operand en in het Destination Function Code register voor de tweede operand. Met behulp van een aparte instructie kunnen deze registers worden beschreven.

Hiermee zijn we wat mij betreft aan het einde van de execution unit. Voor een compleet overzicht van het Supervisor Programmeermodel verwijs ik naar figuur 4. Kijken we nog even naar figuur 1, dan zien we dat alleen het blokje Adress Section nog niet aan bod geweest is. Dit blokje berekent de logische adressen bij de instructies. Het behandelt als het ware de adresseermethodes van de processor. Ik ben van plan deze adresseermethodes in de volgende aflevering te behandelen.

De Memory Management Unit

De 68030 onderscheidt zich van de andere processoren uit de 68000-familie vanwege zijn interne Memory Management Unit. Nu is het niet de bedoeling in deze aflevering alle in's en out's van dit onderdeel te bespreken, nee, daar reserveren we een complete aflevering van deze artikelenreeks voor. Wel wil ik even globaal aangeven wat een MMU

doet en welke registers in het programmeermodel ermee te maken hebben.

Zoals al is aangegeven, is een MMU eigenlijk niet meer dan een tabel waarin logische adressen omgerekend worden in fysieke adressen. Ik begrijp wel dat deze kreet niet veel zegt, daarom zal ik trachten dat door middel van een praktijkvoorbeeld wat duidelijker te maken.

Ik heb begrepen dat Minix voor de 68000 alle programma's zodanig opbouwt dat ze vanaf een vast startadres moeten draaien. Bij een multi-tasking operating system zoals Minix, werkt de processor echter een tijdje (ongeveer 20 milliseconden) aan het een programma waarna hij een interrupt krijgt. Vervolgens kan het zijn dat hij een tijdje (een time slice) gaat werken aan een ander programma. Uiteraard is het dan wel van belang dat beide programma's in het geheugen aanwezig zijn. Als nu een programma alleen kan draaien vanaf een vast adres, geeft dat grote nadelen. Het oude programma moet eerst naar een ander (vrij) stuk geheugen gekopieerd worden zodat het nieuwe programma verplaatst kan worden naar het werkgeheugen. Dat is uitermate tijdrovend. Zou het niet veel gemakkelijker zijn, de adressering in het geheugen zodanig te veranderen dat het startadres nu opeens bij dat nieuwe programma ligt. Welnu, dat is precies wat een MMU doet. In de 68030 kan het geheugen opgebouwd worden uit pagina's van een bepaalde afmeting (minimaal 256 byte, maximaal 32 kB). Van elke pagina wordt het startadres in een datastructuur vastgelegd. Als er nu door de execution unit een logisch adres berekend is, dan bepaalt de MMU in welke pagina het adres ligt en stuurt de adreslijnen

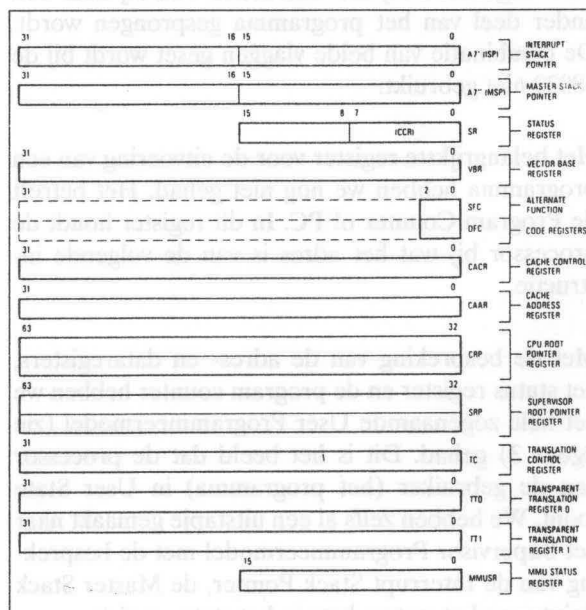


Fig. 4: supervisor programmeermodel

zodanig aan dat de informatie op het fysieke adres benadert wordt. De MMU zorgt er dus voor dat bijvoorbeeld adres \$0101D3F1 omgerekend wordt naar adres \$0B01D3F1.

Behalve dat de MMU de omrekening van logisch naar fysiek adres uitvoert, kan de MMU ook enkele kenmerken van de pagina's vastleggen. Zo kan vastgelegd worden of de pagina benaderd mag worden als de processor in User State is, of er in de pagina geschreven mag worden, en af de pagina cache-able is. Wat dat inhoudt, lees je in de volgende paragraaf.

De opbouw en de werking van de MMU wordt in een aparte aflevering van deze serie uitgelegd. Hier worden ook het Translation Control Register, het MMU Status Register en de beide Root Pointers besproken. De twee Transparent Translation Registers kunnen twee blokken van ten minste 16 MB aangeven waarvoor de MMU buiten werking gesteld wordt. In deze blokken zijn de logische- en fysieke adressen dus identiek. Deze gebieden kunnen bijvoorbeeld gebruikt worden voor de input/output devices van een systeem.

Caches

Binnen een computer zijn er diverse types geheugen. Zo is er bijvoorbeeld het hoofdgeheugen, ten onrechte ook wel met de kreet RAM geheugen aangeduid. Verder zijn er vaak harde schijven en mogelijk zelfs magneetbanden. Ook registers zijn een vorm van geheugen. Nu wil het geval dat zeer snel geheugen meteen ook duur geheugen is. Registers zijn voor de processor zeer snel toegankelijk, magneetbanden juist niet.

In het algemeen is het hoofdgeheugen trager dan de processor. Dit betekent dat het geheugen de processor als het ware afremt. Om dat te ondervangen worden er wel zogenaamde caches toegepast. Dat zijn geheugen-gebieden die sneller zijn dan het hoofdgeheugen. Als er nu informatie (bijvoorbeeld een instructie) uit het hoofdgeheugen gehaald wordt, dan wordt deze meteen ook in een zeer snel stukje geheugen, het zogenaamde cache geheugen, geschreven. Als nu, bijvoorbeeld in een lusje, na een korte tijd hetzelfde adres opnieuw uitgelezen wordt, dan hoeft de informatie niet uit het hoofdgeheugen gehaald te worden maar kan ze uit het cache geheugen gelezen worden. Dit levert in de praktijk een flinke tijdswinst op.

De 68030 heeft op de chip een drietal cache geheugens. In de eerste plaats is dat de zogenaamde ATC (Address Translation Cache) die voor de MMU gebruikt wordt. Hier kunnen 22 entries in

geschreven worden. De twee andere caches zijn een instructie cache van 256 byte en een data cache van 256 byte. De werking van deze caches wordt bestuurd door de twee resterende registers in het programmeermodel, het Cache Control Register en het Cache Address Register. Hoe deze caches precies werken zal ook in een volgende aflevering uitgelegd worden.

Instructies kunnen altijd gecached worden (tenzij iemand het in zijn hoofd haalt door middel van Self Modifying Code het programma te wijzigen terwijl het draait). Bij data ligt dat echter wat anders. Bepaalde geheugen-gebieden mogen niet gecached worden omdat de inhoud van het geheugen kan wijzigen zonder dat de processor daar iets mee te maken heeft. Als voorbeeld noem ik de registers van bijvoorbeeld een DUART die gewoon ergens in het enorme adresbereik van de 68030 een plaatsje toegewezen gekregen hebben. Als iemand op een terminal iets intikt, dan zal de inhoud van deze adressen wijzigen. De informatie die dan eventueel in cache staat, is daarmee verouderd en mag niet meer gebruikt worden. Het geheugengebied waar dergelijke bouwstenen hun registers hebben is dus niet cacheable wat men in de praktijk ook m.b.v. de MMU aan zal geven.

Pipelining

Het laatste waar ik het in deze aflevering over wil hebben is de wijze waarop de 68030 zijn instructies ophaalt. Normaal zou je denken dat de processor een instructie ophaalt, decodeert en uitvoert waarna de volgende instructie aangepakt wordt. Bij de 68030 is het echter al zo dat bij de uitvoering van een instructie de volgende instructie al vast uit het geheugen opgehaald wordt en wordt gedecodeerd. De 68030 voert dit nog wat verder door. De 68030 kan aan maximaal 3 instructies tegelijk werken. Dit maakt de gemiddelde snelheid waarmee de processor een instructie afhandelt aanzienlijk groter dan wanneer hij instructie voor instructie compleet af zou werken.

De techniek waarbij een processor een instructie uitvoert en er al aan enkele andere gewerkt wordt noemt men pipelining. Uiteraard zijn er speciale voorzieningen nodig als de volgende instructie in het geheugen niet de volgende instructie is die uitgevoerd wordt (zoals bijvoorbeeld bij sprongopdrachten). In dat geval is al het werk dat al aan de instructies in de pipeline uitgevoerd is, eigenlijk voor niets geweest zodat de eerste instructie na een sprongopdracht minder snel uitgevoerd wordt. Dit handelt de processor uiteraard zelf allemaal netjes af.

Afsluiting

Zo, dit was de eerste globale excursie door de 68030 processor. Uiteraard is nog lang niet alles verteld; ook niet alles wat nodig is voor een goed begrip van de KGN68k. Ik ben daarom ook van plan nog enkele vervolgen op dit artikel te schrijven waarbij een aantal onderwerpen die nu al aan bod geweest zijn, verder uitgediept worden.

Verder is de verwachting dat de eerste hardware van de KGN68k zo rond november aan de buitenwacht getoond kan worden zodat ook op dat front de resultaten beginnen te komen. Mochten er nog mensen zijn die actief mee willen werken aan de ontwikkeling van de software voor de KGN68k, laat mij of Geert dat dan even weten.

Registratie van shareware

Veel mensen maken gebruik van shareware programmatuur. Nu is de gemiddelde Nederlander niet zo scheutig als het gaat om registratie van shareware en dat is bijzonder betreurenswaardig als je bedenkt dat de shareware industrie zonder registraties niet kan overleven. Het alternatief is dan dat een hele hoop populaire programma's als 4DOS, QuickEdit, Telix, PkZip enz. enz. op commerciële basis voortgezet gaan worden en daarmee onbereikbaar worden voor een grote groep hobbyisten. Deze dreiging is groter dan in eerste instantie lijkt; kijk maar naar de commerciële versies van QuickEdit (OS/2), NDOS (de 4DOS versie die bij de Norton utilities bijgeleverd wordt) en het gebruik van PkZip in commerciële omgevingen. De volgende versie van Telix wordt overigens ook niet meer als shareware uitgebracht...

In het buitenland, en dan zeker in de USA, wordt er voor veel meer shareware betaald in verhouding tot het aantal gebruikers dan in ons kleine kikkerlandje. Een gevolg daarvan is ook dat een aantal programma's in Nederland niet eens meer ondersteund worden. Het wordt dus de hoogste tijd om te gaan betalen voor wat we gebruiken. De KGN wil daarin best een handje helpen. Wie in z'n eentje een licen-

Tenslotte moet ik nog vermelden dat dit artikel mede tot stand gekomen is door de hulp van EBV Elektronik te Maarssenbroek door het beschikbaar stellen van uitgebreide documentatie.

Gert van Opbroek

Literatuur

- 1: Motorola: MC68030 Enhanced 32-bit microprocessor user's manual; Prentice Hall 1990.
- 2: Motorola: M68000 family programmer's reference manual.
- 3: Werner Hilf: Der 68030; MC 4, 5 en 6 1988, Franzis Verlag GmbH.
- 4: Ralf Bermond: Neues Segel gesetzt, Der MC68040; MC 6 1990, Franzis Verlag GmbH.

tie voor een pakket wil hebben betaalt uiteraard meer dan zij die in een groep registreren. Verder beschikt de KGN ook over een credit card, waarmee registreren in het buitenland een stuk eenvoudiger (en goedkoper, en veiliger, en...) is.

Op mijn persoonlijke lijstje aan software dat ik zou willen registreren staan in ieder geval 4DOS, QuickEdit en FastEcho. Die laatste is een uitstekende echomail verwerker voor point-systemen. De andere twee mogen bekend verondersteld worden; ze zijn al eens in "To Share..." aan de orde geweest.

Heeft u ook zo'n lijstje? Of wilt u programma's uit mijn lijstje eens evalueren en eventueel meedoen aan een collectieve registratie? Schroom dan niet om contact op te nemen met ondergetekende. Dat kan per telefoon, elektronische post (points: 2:512/32.2, Internet: voorhaar@utwente.nl), per snail-mail (PTT post), u mag een keertje langskomen of u kunt me aanspreken op één van de clubbijeenkomsten c.q. de HCC-dagen.

Joost Voorhaar

De ene 486 is de andere niet...

Simpel. Dat was het in het begin van het klonengedonder. Er was maar 1 model PC: de PC van IBM. Later gegroeid naar de PC/XT. In die machientjes zat ook een motor. Gemaakt door de General Motors in processorland: ene meneer Intel. Intel had al lang gezien, dat het om twee redenen verstandig was zogenaamde second source overeenkomsten uit te delen:

- Zonder second source ontwerpt niemand met jouw spullen, omdat men altijd een tweede bron wil hebben voor een IC.
- Als het produkt echt aanslaat, kun je zelfs als grote jongen Intel de markt nooit in je eentje voeden: de vraag is eenvoudig veel groter dan de productiecapaciteit.

En zo komt het dat van de 8088 processor uit de PC/(XT) een aantal fabrikanten zijn: behalve Intel maakten ook AMD en NEC 8088 CPU's.

De 8086 (een 8088 met een 16 bit databus) onderging hetzelfde lot, maar die processor is maar in enkele klonen gebruikt. Ook met de volgende generatie 80XXX CPU's, de 80188 en de 80186 ging het minder, want voor zover bekend zijn er geen PC's met dergelijke processors uitgerust.

IBM zat echter net als Intel niet stil en bracht de PC/AT uit, met daarin de 80286 CPU. AMD maakte de processor ook. Harris slaagde er zelfs in, een 80286 in CMOS te maken: ideaal voor portables, of laptops zoals die dingen tegenwoordig heten.

Spelbreker NEC

NEC ondertekende geen second source overeenkomst met Intel. Dat wilden ze best, maar Intel vond dat niet goed. De reden is dat NEC een rottigheidje had uitgehaald: ze hadden de V20 en V30 CPU's bedacht. Compatibel met Intels 8088 en 8086, en tot overmaat van Intels ramp ook nog in staat in een 8080 compatibele mode te werken. Verder waren de V20/30 sneller door een slimmere interne code en architectuur, in CMOS gemaakt (alweer portables!) en goedkoper dan Intels 8088/86 CPU's. Intel bedacht zich dus wel twee keer om NEC een second source overeenkomst voor de 80286 te laten tekenen...

Spelbreker Taiwan

De 80286 begon met een maximum clocksnelheid van 8 MHz, en dat werd via 10, 12 uiteindelijk 16 MHz. AMD deed er nog een schepje bovenop en maakte uiteindelijk zelfs een 25 MHz versie. Ook op andere manieren stak AMD Intel de loef af: men verkocht tegen lagere prijzen. Daarmee was de snelheidshonger van de PC-wereld echter nog niet gestild. In Taiwan werd driftig gedokterd, en het resultaat was een PC/AT compatibel moederbord met daarop niet een 12 MHz 80286, maar een 16 MHz 80386. In het begin waren er nog wel wat problemen, maar die waren na verloop van tijd onder de knie, en daarmee waren de 32-bit moederbord een feit.

Intel was zo langzamerhand beu dat zij wel de ontwikkeling mochten doen, maar niet de gunst van de grote omzet gegund werden. Dus werd met de 80386 een nieuwe weg ingeslagen: Intel zou de enige producent blijven. Dit leidde tot een aantal ontwikkelingen:

- De prijs van de 80386 werd door Intel kunstmatig hoog, en de processor systematisch schaars gehouden.
- Omdat de 80386 duur en schaars bleef, versnelde dit ontwikkeling van

80286 processors met hogere snelheden.

- De Taiwanese, nooit vies van regelrecht jatwerk, zagen een enorme markt voor 386 PC's aan hun neus voorbijgaan en maakten eenvoudig een kopie van de 80386 CPU.

Dit laatste zou een teken aan de wand moeten zijn voor Intel, maar die reageerden niet. De 80386 bleef schaars, en duur.

De volgende stap

Intel bleef zich echter wel bezinnen op het verschijnsel dat zij vrijwel de complete 80286 markt aan vooral AMD verloren hadden. Intel introduceerde iets nieuws: de 80386SX. Dit was intern een complete 80386 32-bit CPU, maar met een 16 bit databus aan de buitenkant. De processor verscheen in eerste instantie uitsluitend als 16 MHz versie. De bedoeling van Intel was namelijk een deel van de 16 en 20 MHz 80286-markt naar zich toe te trekken, door een redelijk geprijsde 16-bit CPU aan te bieden die als voordeel had dat hij 80386 programma's kon draaien. Die waren er toen nauwelijks (Windows 386

De Taiwanese, nooit vies van regelrecht jatwerk, zagen een enorme markt voor 386 PC's aan hun neus voorbijgaan en maakten eenvoudig een kopie van de 80386 CPU.

was 1 van de weinige). Mensen die meer wilden, en de naam Intel op hun CPU wensten waren nog steeds aangewezen op de 32-bit 80386, die inmiddels met zijn complete naam werd aangeduid als 80386DX. Wat later verscheen de 80486.

De 80486DX

Op het eerste gezicht bood de 32-bit 80486DX eigenlijk niets nieuws: het is een combinatie van een 80386 CPU, een 80387 floating point coprocessor en 8k statisch cache geheugen, samengebond op één enorme chip, die meer dan 1,3 miljoen transistoren bevat. Toch was er iets nieuws: de instructie-afwikkeling was sterk verbeterd door aanpassing van de architectuur, zodat bijvoorbeeld een register naar register MOV die bij een 386 3 clocks kost, op een 486 slechts 1 clock duurt. Dat lijkt heel nieuw, naar NEC heeft dit als uitgedaagd met de 8088 versus de V20!

De 486DX verscheen in clockfrequenties van 25 en 33 MHz, en wat later ook in 50 MHz. Die laatste processor stond tot voor kort te boek als de snelste commercieel verkrijgbare microprocessor. Een goed ontworpen PC met deze CPU geeft een Norton SI-factor van meer dan 80 te zien! Een nadeel is en blijft de prijs: ook hier slaat Intel toe en houdt prijs hoog en de processor relatief schaars. Ook nu geen second source overeenkomsten.

Spelbreker AMD

En dat zal AMD niet lekker. Intel had de complete 80386 en 80486 markt in handen, en die markt kon belangrijk groeien als de prijs naar beneden zou gaan. AMD trok de stoute schoenen aan. Het verhaal wil, dat een groot aantal Intel 80386DX processors atoomlaag voor atoomlaag door AMD werden 'geschild' en gefotografeerd onder een elektronen-microscop. Na vele maanden zwoegen was AMD achter de opbouw en geheimen van de Intel 80386DX. Het werk kon beginnen: een processor ontwerpen die intern duidelijk anders is, maar aan de buitenkant zich precies zo gedraagt als een Intel 80386DX. Dat lukte. Dat lukte zo goed, dat Intel en AMD elkaar juridisch zeer uitgebreid in de haren vlogen. Het resultaat was afgelopen winter dat AMD zijn eigen Am386 (want zo heet 'ie) toch mocht blijven verkopen. En passant maakte AMD nog even een 40 MHz versie die Intel niet in het programma had. Het resultaat is nu duidelijk zichtbaar: 80386DX moederbord zijn inmiddels uiterst betaalbaar geworden.

Het gerucht wil, dat AMD op dit moment werkt aan een kloon van de Intel 80486DX CPU. We zullen het vanzelf wel merken.

Zelfs IBM doet mee

De 80386SX was een leuke CPU, maar in de ogen van sommigen te langzaam, terwijl er prijstechnisch een groot gat tussen de 80386SX en de 80386DX zat. IBM loste dit op door netjes in overleg met Intel de 80386SX op te voeren naar 20 MHz, en te voorzien van een 8kbyte interne cache. IBM heeft kennelijk aan Intel moeten beloven de chip uitsluitend voor eigen gebruik te benutten, want de 80386SLC is niet los te koop, maar alleen in de PS/2 Model 57SLC.

Nog meer 80386's

De truc met de on-chip cache werd door nog iemand bedacht: Cyrix. Inmiddels geen onbekende in CPU-land, want Cyrix heeft met haar coprocessoren de markt voor Intel op dit gebied grondig verziekt.

Ook hier met dezelfde truc: lagere prijzen en een iets andere architectuur dan het origineel. Cyrix nam de 80386SX onder handen, bouwde de microcode om zodat een aantal instructies sneller verliepen, en monteerte een 1 kbyte cache op de chip. De naam werd... Cx486SLC. En daarmee is het begin van de verwarring daar: de Cx486SLC is een 80386!

Ook Chips&Technologies, wereldwijd beroemd om haar chipsets voor moederborden, heeft klonen

van de 80386DX en 80386SX gemaakt die 38600DX en 38600SX heten. Deze chips zijn sneller dan de Intel voorbeelden, maar wel pin-to-pin compatibel. De 38600DX is thans mondjesmaat verkrijgbaar. Hoogste clocksnelheid: 40 MHz. C&T werkt nu al aan opvolgers: de 38605DX/SX die nog sneller moeten worden.

Intels antwoord

Intel blijft bij als dit geweld doorvechten en komt met een aantal antwoorden. Het eerste is het antwoord op AMD's 386DX-40: de 80486SX. Ha, denkt iedereen, dat is dus de 16-bit 80486. Mis poes! De 80486SX is een 80486DX, dus 32-bit, maar zonder floating point coprocessor en is dus even snel als een DX, zolang er geen floating point coprocessor nodig is. Hoogste snelheid 80486SX die wordt aangeboden is 25 MHz. De SX past in de voet van de DX. Nu is het nog duidelijk. Maar er is meer. Er is ook een 80487SX. Ha, denkt iedereen nu: dat is dus de floating point coprocessor die bij de 80486SX ontbreekt. Weer mis. De 80487SX is een complete

Ook hier slaat Intel toe en houdt prijs hoog en de processor relatief schaars.

80486DX, maar bedoeld voor de coprocessorsocket die Intel heeft gespecificeerd voor alle 80486SX moederborden. De 80487SX zet namelijk de op het moederbord aanwezige 80486SX compleet buitenspel. De reden voor dit mafs is, dat de coprocessor verweven is met de gewone CPU en niet via een gebruikelijke busstructuur met de gewone CPU communiceert. Gevold van dit al is een kind met een technisch waterhoofd, en dat de combinatie van 80486SX+80487SX duurder is dan een echte 80486DX.

Als u dacht dat we er nu waren dan heeft u het mis. Er is inmiddels ook een 80486DX2. Dat is een 80486DX die intern met dubbele clock werkt, of iets anders uitgedrukt: de CPU doet iets op iedere clockflank. Maximum snelheid op dit moment is 66 MHz en daarmee is de 80486DX2-66 de absolute top.

We zijn er nog niet: Wat dacht u van RapidCAD? Dat is een 80486DX, bedoeld voor de coprocessorsocket van 80386DX systemen. Vanwege de busstructuur komt de performanceverbetering echter nauwelijks uit de verf. Intel voorziet vooral verbetering bij coprocessor-intensieve toepassingen, zoals CAD-software, vandaar de naam.

Nog is de koker met ideeën bij Intel niet leeg: er zijn ook nog overdrive 486's. Dat zijn ook dubbelclockers, maar nu voor een bestaande 486DX. Op dit moment alleen voor 486SX (maximaal 25 MHz), voor een prijs waarvoor je ook bijna een 50 MHz 80486DX moederbord koopt. De Overdrive DX moet nog verschijnen.

Verwarring?

Met al die 486 varianten maakt Intel het voor de gebruiker beslist niet duidelijker, zoals hiervoor reeds is gebleken. Cyrix doet er nog een schepje bovenop door een 80386-variant 486SLC te noemen. Daarom hier een opsomming die naar ik hoop enige duidelijkheid brengt:

Intel 80386DX/AM386DX

32-bit processor, 20 byte adres pijplijn, maximum snelheid 33 MHz (Intel) of 40 MHz (AMD).

Intel 80386SX

16-bit databus versie van de Intel 80386DX. Maximum snelheid 20 MHz.

C&T38600DX

32-bit processor, snellere interne verwerking dan Intel 80386DX, maximum snelheid 40 MHz. Verschijnt misschien ook in 16-bit (SX) vorm.

IBM 80386SLC

Intel 80386SX met 8k on-chip cache. 20 MHz maximum. Niet los verkrijgbaar.

Cyrix 486SLC/DLC

16- resp. 32-bit processors, compatibel met, doch sneller dan Intel 80386SX/DX, 1k on-chip cache, maximum 25 MHz (SLC) of 40 MHz (DLC).

Intel 80486DX

32-bit processor, floating point processor en 8k cache on-chip, sneller dan 80386DX bij dezelfde clocksnelheid. Maximum 50 MHz.

Intel 80486SX

80486DX zonder floating point processor, doch met 8k cache. Maximum 25 MHz.

Intel 80486DX2

80486DX met intern dubbele clock. Maximum snelheid 66 MHz. Niet los verkrijgbaar.

Intel 80487SX

80486DX bedoeld voor de coprocessorsocket van 80486SX systemen. Maximum 25 MHz.

Intel Overdrive SX

Vervanger voor 80486SX met dubbele interne clock. Maximum 25Mhz (=performance van de niet bestaande 50 MHz 80486SX)

Intel Overdrive DX

Vervanger voor 80486DX met dubbele interne clock. 20- Mhz versie aangekondigd. Performance is dan gelijk aan 80486DX-40.

Intel RapidCAD

80486DX bedoeld voor de 80387DX socket van 80386DX systemen. Levert een slechts magere performanceverbetering. 33 MHz maximum.

Conclusie

De top: Intel 80486DX2-66. Goede keus nu: Am386DX-40. Daartussen in: Intel 80486DX-50. Geen floating point: Intel Overdrive SX. Wat langzamer: Cyrix 486SLC. Niet aanbevolen: Intel RapidCAD (Am386DX met Cyrix 387 coprocessor is sneller en goedkoper). Grote verliezer: Intel 80386DX. Komende kraker: Am486DX op 66 MHz, wellicht met dubbele clock. Algemeen: verwarring alom.

Aanbeveling aan de heren CPU-bouwers: houd het duidelijk. Netto resultaat voor de eindgebruiker: als er geen Intel op staat is het goedkoop en sneller. Staat er wel Intel op, dan staat er een 4 in het midden, is het dubbel zo duur en anderhalf maal zo snel. Voorlopig doe ik het op een Intel 80486DX-25. Naar mijn maatstaven loopt dat al hartinfarctverwekkend hard...

Nico de Vries

Nieuwtjes en andere wetenswaardigheden

Tijd voor een nieuwe PC...?

Nu, en in de komende maanden is het misschien het juiste moment om eens naar een nieuwe PC om te gaan kijken. Waarom nu? Wel in ons lokaal roddelblad stond een artikel dat mij aanzette om eens goed om me heen te kijken. Volgens dat blad was er een oorlog uitgebroken. Nu eens niet een oorlog die je de haren ten berge doen rijzen. Maar een oorlog waarvan je als gewoon burger beter kunt worden. De slachtoffers zijn dit keer de producenten en handelaren van ons troetelkindje, de computer. Een tulpen producent uit Den Bosch had de pers laten weten dat ze de prijzen met 20 tot 30 procent zouden verlagen. Een bittenbakfabriek in Enschede zit in de moeilijkheden. Kleine leveranciers komen één voor één in de problemen, omdat de prijs waarvoor ze inkopen een paar dagen later de verkoopprijs is bij de concurrent. Een concurrent die vaak vanuit een grote inkoopcombinatie werkt. En dan niet alleen uit ons kleine kikkerlandje, ze schijnen tegenwoordig ook al uit Duitsland te komen. Dat kan nog wat worden als die grenzen ook op papier open gaan. Het is nu al moeilijk voor mij om boodschappen te doen in de stad omdat er een file staat met hunkerende Duitsers. Als ze dadelijk ook nog computers komen verkopen, moet ik maar boodschappen gaan doen in Duitsland.

.... maar dan zonder goedgekeurd modem.

Volgens een test van een van onze consumentenorganisaties bestaan er geen modems die aan de wettelijke toelatingseisen voldoen. De hoogste waardering die ze wilden toekennen was: matig. Dat geldt alleen als je bent aangesloten op een toonkiescentrale. Met andere woorden er wordt ons door de overheid en de PTT weer eens onnodig geld uit de zak geklopt. Zelfs het door de Postbank geleverde externe-modem voor Girotel-abonnees heeft in de meegelerde netadapter een fout. En met dat zootje ongeregeld wordt je dan ook nog eens aangemoedigd om gebruik te maken van videotex-banken.

Volgens dezelfde test is het daar al niet veel beter gesteld als met de modems. Het is daar slecht zoeken, veel beloven en weinig vinden. Behalve als je bereid bent om lang online te blijven. Maar dan rinkelt de kassa weer bij de PTT.

Eén oorlog is binnenkort voorbij.

Als je de gebruikers van een Apple of IBM-PC met elkaar hoort discussiëren dan vliegen de stukken je soms om de oren. Ik denk dat deze heftige discussies hun langste tijd hebben gehad. IBM, Motorola en Apple hebben samen een design en development afdeling opgericht in Austin, Texas. Als deze samenwerking zijn vruchten begint af te werpen komt er vanzelf een eind aan voornoemde discussies.

Hard disk ter grootte van een lucifers doosje.

De grootte van de hard disk gaat zienderogen achteruit. Niet dat dat nu meteen een probleem is maar het gaat wel erg hard. Toen de 5 1/4" hard disks uit kwam spraken we onze bewondering uit. Mooi klein en compact. We hadden het nauwelijks uitgesproken of we werden overspoeld met 3 1/2" hard disks. Hewlett-Packard wil ons binnenkort toch weer in verbazing doen uitbarsten. Ze zijn deze maand begon-

nen met de productie van de Kittyhawk. Een hard disk ter grootte van 1.3". De Kittyhawk heeft een opslag capaciteit van 21 MB en heeft een gemiddelde zoektijd van 18 ms. Het prijsje valt reuze mee, als er tenminste 1000 afneemt, dan kost hij slechts \$ 210, = =.

Klik MS-DOS en klik UNIX

Opus Systems heeft een kaart ontwikkeld die schakelt met een klik van de muis of met een toetsaanslag tussen MS-DOS of UNIX. De kaart is gebaseerd op een 40 MHz SPARC-processor. Het prijsje zal nog niet voor elke hobby-ist aantrekkelijk zijn, die is vanaf \$ 4145, = =.

Er is een oorlog uitgebroken. Nu eens niet een oorlog die je de haren ten berge doen rijzen.

Connect 2400

Modems zijn aardige speelgoedjes. Helaas heeft niet iedereen de mogelijkheden en/of financiële middelen om intensief gebruik te maken van een dergelijk apparaat. Het BBS begint echter een steeds belangrijker plaats in de club in te nemen (naar mijn mening geheel terecht overigens, maar wellicht is mijn visie ietwat "gekleurd" te noemen). Als een soort spin in zijn web is "The Ultimate" verbonden met een heel scala aan andere computers waarmee in de nachtelijke uurtjes geheel automatisch informatie uitgewisseld wordt. In die grote, dikke informatiebrei gaan soms hele aardige verhandelingen, discussies en (nieuws-) feiten schuil. De "echt interessante" berichten en/of threads bewaar ik meestal in een speciale archive, en het lijkt me een aardig idee zo af en toe eens wat te publiceren uit dit elektronisch archief. Onder het kopje "Connect 2400" zult u in de toekomst dan ook regelmatig wat van die informatie terug kunnen vinden. Daar de meeste interessante messages behoorlijk lang zijn zal ik ze meestal niet in hun originele vorm laten staan, maar ze verknippen en/of geheel herschrijven.

Verschillende netwerken

Zoals gezegd maakt het bulletin board deel uit van een aantal verschillende netwerken. Deze netwerken verdelen we grofweg in twee groepen: netwerken voor de verspreiding van elektronische post en netwerken voor de verspreiding van files. Ofschoon de laatste groep in de meerderheid is zal ik me in deze rubriek beperken tot de (bijna) puur op elektronische postuitwisseling gerichte communicatie.

Elektronische post kan ook weer verdeeld worden in een aantal vormen. Al naar gelang het netwerk wordt er gesproken over "netmail" versus "echomail" of over "E-mail" versus "news". Netmail is persoonlijke post, en wordt van de originele verzender zo direct mogelijk op de plaats van bestemming gebracht. In de praktijk is netmail altijd privé post. Echomail daarentegen wordt verspreid over een groot aantal andere systemen en zal vrijwel altijd een "open" karakter hebben. Dat wil dus zeggen dat in principe iedereen die op een aangesloten systeem inlogt die berichten kan lezen en er op kan antwoorden. Echomail (of news) is een uiterst open vorm van communiceren en zal de bron vormen voor het grootste deel van de berichten in "Connect 2400".

De meest waardevolle berichten (in ieder geval voor deze rubriek!) komen op "The Ultimate" binnen via

het fidonet of via de Internet aansluiting. Fidonet is een wereldwijd netwerk van PC-achtige systemen en richt zich in de eerste plaats op de hobby-sfeer. De meeste nodes in fidonet draaien dan ook op privé systemen van mensen die hun vrije tijd voor een groot deel in het netwerk gebeuren steken. Internet is een professioneel netwerk dat zich in eerste instantie richt op universiteiten en research & development afdelingen van commerciële instellingen. De Internet nodes ("sites" genoemd) draaien op machines variërend van PC-achtigen tot mainframes, van MS-DOS en UNIX tot VMS en wie weet wat allemaal nog meer voor systemen. De meeste berichten ten aanzien van UNIX komen van het Internet, de overige berichten komen meestal van het fidonet.

Areas en newsgroups

Alle echomail en news wordt in verschillende gebieden opgedeeld. Zo'n gebied wordt in fidonet aangeduid als "area", op het Internet noemt men zo'n gebied een "newsgroup". Alle area's hebben een unieke naam waarmee het onderwerp van de area aangeduid wordt. Zo heet de internationale area over shareware bijvoorbeeld "SHAREWRE" en de heeft newsgroup over Minix de naam "comp.os.minix" gekregen. Van alle publicaties in "Connect 2400" zal ik, indien mogelijk, deze namen én het onderwerp waaron-

der het betreffende bericht (de berichten) gecatalogiseerd is (zijn) vermelden.

Genoeg inleiding lijkt me zo... let's get to the real thing!

Muizen

Fred van Kempen, welbekend binnen de KGN als (voormalig) voortrekker van de NL-MUG, lanceerde in comp.os.minix een mousedriver voor PC-Minix ("PowerMOUSE: A mouse driver for MINIX"). Jawel, een stukje programmatuur dus. Dit is typisch voor Internet overigens; in fidonet wordt het meestal niet op prijs gesteld om programmatuur en/of listings in de echomail te droppen. De driver zal binnenkort ook op The Ultimate beschikbaar zijn in de Minix support area.

Minix vanuit DOS

Een ander stukje software komt van Tsung-Fan Li. Hij postte (wederom in comp.os.minix) onder de titel "Help on DOS started Minix boot" als antwoord op een vraag om hulp een programma waar-

mee het mogelijk is om Minix vanuit MS-DOS op te starten. Daarbij is zo op het eerste gezicht geen Minix-bootdisk meer nodig, maar kan volstaan worden met een image-file onder DOS. Ik heb er nog niet mee geëxperimenteerd, maar ik zal de listing wederom in de Minix support area opnemen.

Shareware Industrial Awards

Van fidonet kwam ik in de SHAREWRE area een bericht tegen waarin ene Richard Holler melding maakt van de eerste uitreiking van de "Shareware Industrial Awards". Dit moet een jaarlijks terugkerende prijs worden voor de beste shareware producten. In de categorie "beste utility" werden 4DOS, IconMaster, Vern Buerge's List, PKZip en ZipKey genomineerd. De prijs ging uiteindelijk naar de makers van 4DOS. Verder was er een categorie "beste applicatie", waarin PC-File, PC-Write, ProComm, Qedit, Telix en "As-Easy-As" genomineerd werden. De auteurs van "As-Easy-As" gingen er met de eerste prijs vandoor. De prijs voor de beste grafische tools en applicaties ging naar het pakket "Paint Shop Pro" van Robert Voit. Overige genomineerden waren "Cooper Graphics Series", "DeskTop Paint", "Draft Choice" en "Graphics Workshop".

Erg klein was blijkbaar de categorie "Best Programming Languages and Tools"; er waren slechts drie genomineerden waarvan A86 van Eric Isaacson de prijs in de wacht sleepte. De andere genomineerden waren de "TechonoJocks Turbo and Object Toolbox" van Bob Ainsbury en "Window Boss and Data Clerk" van Philip Mongelluzzo.

Verder waren er prijzen voor de beste zakelijke financiële pakketten (gewonnen door het Medlin Accounting System), educatieve software (Animated Series van Tom Guthery IV), en de beste "onderhoudende" software (spelletjes dus...). De laatste werd gewonnen door "Commander Keen" van Apogee software, die ook "overall winner" werd. Tenslotte waren er nog prijzen voor de categorieën "Best Home or Hobby Software", "Best Math or Engineering", "Best New Product" en er was een speciale publieksprijs. Deze publieksprijs werd aan J.P. Software toegekend voor hun welbekende "4DOS".

Een aantal van de genomineerde en/of prijswinnende programma's zullen we, voorzover dat nog niet gebeurd is, bekijken in het kader van de rubriek "To Share...".

De Minix kernel

Het laatste nieuws komt wederom van comp.os.minix. Frans Meulenbroek, al enige jaren een bekende verschijning in de Minix-wereld, stelde

een radicale verandering in de Minix kernel voor onder het kopje "minix kernel structures". Naar aanleiding van de ingewikkelde reeks handelingen die benodigd zijn om nieuwe drivers aan het systeem toe te voegen bedacht hij een systeem waarmee drivers als externe tasks aan een micro-kernel toegevoegd worden. Op deze manier is het bijvoorbeeld niet meer nodig om het hele operating systeem opnieuw te compileren zoals nu het geval is. De taken van de microkernel omschrijft hij zelf als volgt:

This core should have the following functions:

- context switching
- message passing
- basic clock handling
- memory copying between tasks
- support for interrupt handling
- creation and removal of processes.
- system call dispatching (perhaps this should be done by a special task)

Functions supplied by the core itself are:

- the functions from system.c and proc.c
- install interrupt handler
- deinstall interrupt handler
- install sytem call (allowing a process to register itself as servicing source for a specific system call)
- attach to major device number (this way device numbers can be connected to drivers). This can simply be the mapping of the device number to the task id.

Helemaal uitgewerkt is het voorstel nog niet, want hij heeft nog geen oplossingen voor het booten van het hele zaakje en voor de plaats waar hardwareinterrupts verwerkt zouden moeten worden. De voordelen van zijn oplossing zijn duidelijk: er ontstaat een compactere kernel die flexibeler en overzichtelijker is dan de huidige kernel. Een nadeel vindt hij de toename in omvang van de code en een klein stukje verlies aan snelheid. Als je zijn verhaal en de opvolgende antwoorden leest, krijg je trouwens sterk de indruk dat Frans wat ideeën opgepikt heeft van OS/9.

Zo, dat was het wel zo'n beetje. Mogelijk lekt tussen de regels wat van mijn enthousiasme voor deze moderne vorm van communiceren door naar u, de lezer. Misschien zelfs dat u er door aangestoken wordt en eens rondbladert in de echomail op The Ultimate. Te veel gebruikers komen slechts op het BBS voor de rijke verzameling aan bestanden, downloaden wat van hun gading is en kijken eigenlijk niet verder. En dat is dus best jammer!

Joost Voorhaar

Voortgang KGN68k

Jawel, we zijn de zomer door en er is nog steeds sprake van voortgang in het KGN68k project. Misschien wel meer dan ooit van te voren. Ik zou zelfs van een stroomversnelling willen spreken.

Printed Circuit Board

De printplaat is een tijd lang het grootste probleem geweest. Het formaat, de prijs van het prototype en de capaciteit van het printontwerppakket waren de grootste struikelblokken. Een echt knelpunt, reden waarom enkele leden van de werkgroep zelfs wilden afhaken. Er is toen besloten om een stap opzij te maken, we hebben het moederbord verdeeld over verschillende printplaten van het standaard format 160 bij 100 millimeter (EUROCARD). U leest het goed: weer is het mechanische concept veranderd. In eerste instantie was er wel wat tegenstand om de zaak wéér om te gooien. Door de diverse voordelen gaan we hier echter toch mee verder; de overstap naar motherboard formaat is in een later stadium altijd nog mogelijk.

Die voordelen van het eurcard concept zijn legio: we blijven per print ver onder de grens van PCB ontwerp pakket; meerdere personen kunnen te gelijk

aan het layouten zijn; en (last but not least) alleen de processor print hoeft maar een dure vierlaags print te worden. Een nadeel is dat het lastig zal zijn om PC-insteekkaarten mechanisch te plaatsen.

Bus

Alle kaarten hebben een haakse DIN41612ABC mannetjes connector. Het backpanel heeft dus vrouwtjes connectors, waarvan alle contacten 1:1 verbonden zijn. De buitenste zes pinnen worden gebruikt voor de 5 volt voedingsspanning. Door de recht-toe-recht-aan aansluitingen op de busprint was het niet mogelijk

om een daisy chain interrupt mechanische te gebruiken. Hoe de pin-out er dan nu uit ziet is te zien in figuur 1.

De adres & data lijnen zitten alles behalve op een rijtje; deze 62 lijnen moesten toch aan de buffers en aan de 96-polige connector vast zitten. Door geen volgorde aan te houden was het mogelijk om één en ander te realiseren.

Kaarten

Het ontwikkelpad van de diverse kaarten is als volgt:

processorkaart, I/O-kaart met EPROM en SRAM, diskkaart, DRAMkaart en dan het AT brigde board. De eerste drie kaarten zitten direct op de busprint en de DRAMkaart zit als piggy op de processorkaart. Van de eerste kaart wordt nu de lay-out gecontroleerd en kan dan naar de printboer. De IOkaart met het 8-bit geheugen nadert ook al ras de voltooiing van de lay-out. Die twee kaarten en een busprint willen we

tijdens de komende HCC-dagen 'Hello, world' laten zeggen.

Software

De vorige keer werd er gesproken over een assembler waar ijverig aan gebouwd werd. Nu, die assembler genereert al code die op een Atari ST draait. Samen met dat gereedschap word een ander stuk gereedschap, de monitor, gemaakt. Andere ontwikkelingen gemaakt met de KGN68k assembler zijn elders te vinden in dit blad over (trace) exceptions van de Motorola 68000. Ook wordt er met de GNU 68k compiler gesleuteld; wat ik heb begrepen heb is dat het linken nog niet helemaal naar wens gaat.

Echte tijger

Ongeveer anderhalf jaar lang is de KGN68k een papieren tijger geweest. Nu zijn we dan in een stroomversnelling geraakt. We hebben zo ongeveer de eerste printen in handen en diverse grote stukken van de software puzzel. De tot nu besteedde tijd beschouw ik dan ook als het leggen van een stevig fundament. Het winnende team dat nu de KGN68k-werkgroep is wil ik niet veranderen, maar als U het aan de software kant wilt versterken of steunen kunt U contact met me opnemen.

Geert Stappers

PIN	A	B	C
1	GND	GND	GND
2	-12V	+12V	BA0
3	BA28	BA29	BA1
4	BA26	BA27	BA24
5	BA25	BA23	BA22
6	BA21	BA20	BA18
7	BA19	BA16	BA17
8	BA14	BA11	BA15
9	BA13	BA9	BA12
10	BA5	BA8	BA10
11	BA6	BA4	BA7
12	BLACK	BA2	BA3
13	BIRQ1	BAS	BSIZ0
14	BIRQ2	BDS	BSIZ1
15	BIRQ3	BDSACK0	BR/W
16	BIRQ4	BDSACK1	BFC0
17	BIRQ5	BBGACK	BFC1
18	BIRQ6	BRESET	BFC2
19	BIRQ7	BBG	CLK16
20	BBERR	BCSIO	NC
21	BBR	BD0	BD1
22	BD2	BD4	BD5
23	BD3	BD6	BD7
24	BD8	BD11	BD9
25	BD13	BD10	BD12
26	BD15	BD14	BD16
27	BD17	BD18	BD20
28	BD19	BD22	BD21
29	BD24	BD23	BD30
30	BD27	BD25	BD29

Fig. 1: pinout van de busconnector

Altijd de juiste tijd

Tijd speelt in het moderne leven een grote rol. Afspraken worden op basis van tijd gemaakt en wie te laat is loopt het risico een vergaande relatie stuk te laten lopen...

In 1978 is de eenheid van tijd, de seconde, vastgelegd als de tijd die benodigd is voor bepaald aantal wisselingen in energieniveau's van het atoom cesium. Op basis van deze definitie heeft men in Duitsland bij het Physikalisch Technische Bundesanstalt een atoomklok gebouwd die als meest nauwkeurige uurwerk in de wereld gezien wordt. Deze klok is verbonden met een zender in de buurt van het Duitse Mainflingen (bij Frankfurt am Main in de buurt). Deze zender zendt met een vermogen van ca. 25 kW een carrier uit op 77,5 kHz. Deze carrier wordt aan het begin van iedere seconde teruggebracht tot zo'n 25% van de maximale amplitude. Na 100 of 200 milliseconden wordt het signaal weer op volle sterkte uitgezonden. Op deze manier wordt er een digitaal signaal met de gigantische snelheid van 1 bit per seconde de ether ingeslingerd. Wordt de carrier 100 milliseconden lang gereduceerd in amplitude, dan hebben we een 0 te pakken; 200 milliseconden betekent een 1.

Met behulp van een eenvoudige ontvanger kan deze stroom bitjes omgezet worden naar digitaal interpreteerbare signalen. Een uitstekende bouwsteen hiervoor is bijvoorbeeld de DCF-77 ontvanger van

Afspraken worden op basis van tijd gemaakt en wie te laat is loopt het risico een vergaande relatie stuk te laten lopen...

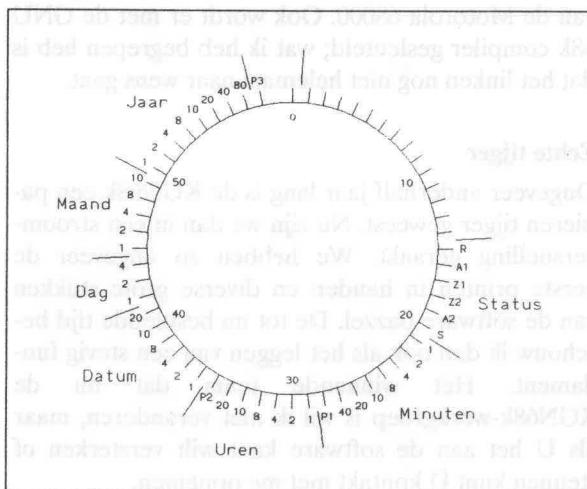
Conrad Electronic. Dit is een lekker compacte ontvanger in SMD-technologie met ongeveer het formaat van een luciferdoosje. Dit ontvangertje maakt onderdeel uit van een complete insteekkaart voor de PC, maar er is ook een (veel goedkopere) uitvoering voor de Commodore C64. De laatste is natuurlijk niet voorzien van een PC-compatible insteekkaart, maar het ontvangertje is precies dezelfde. Met een eenvoudige externe voeding (5V, 3 mA) kan de module op een printerpoort aangesloten worden. Het ding kan natuurlijk ook aan een eventuele gameport gekoppeld worden (de voeding kan dan uit diezelfde gameport getrokken worden), maar dan kan de DCF niet onder interrupt uitgelezen worden. Het

schema van deze fantastisch ingewikkelde schakeling (ahum...) zullen we deze keer maar achterwege laten... Het binnenkomende DCF-signaal bestaat uit een stroom van 59 bitjes per minuut. Het zestigste bit ontbreekt, waardoor het mogelijk is het begin van een nieuwe minuut te detecteren. De codering van het systeem wordt vaak met behulp van figuur 1 weergegeven. De eerste 15 bits bevatten geen informatie; in de praktijk worden er nullen uitgezonden. Vervolgens komen er achtereenvolgens: het

reserve-antennebit, een aankondigingsbit voor een wisseling in zomer- en wintertijd, de zomer- en wintertijdbits, een aankondigingsbit voor een correctieseconde en een startbit. Die correctieseconde wordt zo af en toe toegevoegd (als 61ste seconde in een minuut dus!) om de klok te corrigeren voor afwijkingen in de omwentelingssnelheid van de aarde. Na deze verzameling volgt de tijdinformatie in BCD, beginnend met de minuten. Vervolgens de minuten, de kalenderdag (datum), de weekdag (1..7, 1 = Maandag), de kalendermaand en tenslotte het kalenderjaar. Een drietal parity bits (P1 t/m P3) complementeren het signaal.

Tenslotte volgt nog een simpel programma om de DCF clock mee uit te kunnen lezen. Dit programma is meer als uitgangspunt bedoeld voor verdere experimenten dan als werkelijk toepasbare klok. Zo ben ik momenteel bezig om een resident programma te maken die de klok uitleest en het resulterende signaal als RTC beschikbaar stelt. Veel plezier ermee!

Joost Voorhaar



Figuur 1: De DCF klok

KG68k ISA busprintdefinitie

Doel van de ISA busprint

Het doel van de ISA busprint is vrij eenvoudig te omschrijven. De ISA bus print heeft tot doel een koppeling tot stand te brengen tussen een (geïsoleerde) ISA bus en de KG68k computer.

Omdat verschillende mensen graag nog ISA bus kaarten in deze computer willen plaatsen is besloten om in de KG68k een zelfstandige ISA bus te plaatsen. De hier beschreven busprint zal dan de koppeling tussen de KG68k (bus) en de ISA bus voor zijn rekening nemen.

Globale definitie van de te ondersteunen ISA busfuncties:

De ISA busuitbreiding heeft geen volledige ISA bus ondersteuning. Om de print niet te ingewikkeld te maken zijn slechts de volgende functies aanwezig:

- 8-bit en 16-bit breed geheugen.
- 8-bit en 16-bit breed I/O.
- Afhandeling van de standaard interrupts.

Hierbij moet wel het volgende worden opgemerkt. De I/O adresruimte is 16 bits breed. De geheugen adresruimte is 24 bits breed. Het stuk adresruimte dat in gebruik is door I/O is afgehaald van het stuk geheugen. Voorlopig is de geheugenopbouw tussen

de KG computer en de ISA bus als volgt gedefinieerd:

\$C0000000 t/m \$C000FFFF:

ISA I/O adresruimte: \$0000-\$FFFF

\$C0010000 t/m \$C0FFFFFF

ISA geheugenadresruimte: \$010000-\$FFFFFF

Dit zal de praktijk geen probleem opleveren omdat de onderste 64k geheugen in nagenoeg elke PC\XT\AT enzovoorts al bezet is door eigen geheugen en er dus geen uitbreidingen zijn die specifiek in dit stuk geheugen moeten.

De DMA en busmaster eigenschappen van de ISA bus zijn niet op deze kaart aanwezig. Mocht hier genoeg belangstelling voor bestaan dan zal een aanpassing of uitbreiding van deze kaart worden overwogen. Deze reden om deze twee functies niet op te nemen is behalve de moeilijkheidsgraad ook nog de snelheid. Bovenstaande functies zijn speciaal voor snelle data overdracht. Wie dit wil verwezenlijken kan beter een 32-bits kaart maken voor de KG68k bus.

Verder is de timing niet geheel identiek aan die van de ISA bus. De verwachting is dat minimaal negentig procent van de kaarten probleemloos kan worden gebruikt in samenwerking met de ISA bus print.

Hugo van der Kooij

We hebben een ridder in ons midden

Donderdag 27 augustus togen 2 van jullie bestuursleden naar Krommenie. We hadden een uitnodiging ontvangen om aanwezig te zijn bij het jubileum van Co Filmer. Co maakt al 40 jaar bij Forbo de omgeving onveilig met zijn electrotechnische hersenpinsels. Nu is het niet de gewoonte om naar al dat soort festiviteiten heen te gaan, maar voor een lid dat al vanaf de oprichting actief is wilden we een uitzondering maken.

Nou, dat hebben we geweten. De bezoekers van de bijeenkomsten in Krommenie kennen de kantine/ontvangst-ruimte van Forbo. Die zat niet goed gevuld maar gewoon vol. Zoals gewoonlijk op dat soort feestjes werd de jubilaris en zijn familie geprezen, geprezen en nog eens geprezen. Maar ook de KIM-club bleef niet onvermeld. Alle sprekers noemden die drie letters wel een keer in hun betoog. Ik voelde me daardoor zo uitgelaten dat ik voor de zekerheid de veters in m'n schoenen wat strakker heb aangetrokken. Je zal daar ineens van verwaandheid naast je schoenen gaan lopen. Dat kon ik in aanwezigheid van onze oprichters Siep de Vries, zijn vrouw en Anton Müller natuurlijk niet maken. Rinus Vleesch Dubois moest om gezondheidsredenen he-

laas verstek laten gaan.

Het bleef niet alleen bij mooie woorden. Co werd overladen met cadeaus en bloemen, ook voor zijn lieve vrouw. Dat had ze natuurlijk wel verdiend, want zij heeft er ook toe bijgedragen dat hij het daar 40 jaar heeft volgehouden. Tussen neus en lippen door werd Co nog even door de burgermoeder op het podium gevraagd. Zij vond dat hij zo goed z'n best had gedaan dat het nodig was om onze Co tot ridder te slaan. Die klap kwam zo hard aan dat Co er even stil van werd. Zo stil dat hij bijna geen woorden meer had om zijn dank uit te spreken. Volgens ingewijden neemt hij voor dat soort betoogjes geen minuten maar kwartieren. Gelukkig bleef het dit keer bij 2 minuten. Mogelijk dat hij zijn directeur al wilde laten voelen dat zijn verzoek aan Co, om als hij over 5 jaar met de VUT gaat toch stand-by te blijven, ook van korte duur zal zijn.

Co, namens de leden willen we jou en je familie nogmaals feliciteren met je jubileum en je verdiende ridderschap.

Tonny Schäffer

Gids voor electro-technici

- 1 Electriciteit wordt gemaakt in een fabriek, waar het in kabels wordt opgeslagen. Daarom zijn de kabels zo lang. Soms worden ze op haspels gewonden om meer kiloWatt te krijgen.
- 2 Sommige soorten electriciteit maken geen gebruik van kabels, zoals bij onweer en in draagbare radio toestellen. Deze soorten worden niet opgewekt, maar worden in de lucht aangetroffen.
- 3 Electriciteit maakt een laag zoemend geluid. Dit kan worden gebruikt voor deurbellen, telefoneren en elektrische orgels. Men kan er ook de buurman mee pesten, hoewel dat enige studie vereist.
- 4 Electriciteit moet geaard worden, behalve bij vliegtuigen.
- 5 Electriciteit wordt gemaakt uit twee ingrediënten, negatief en positief. Een ingrediënt heeft de voorkeur voor rode draden, de andere voor zwarte. Wanneer deze draden elkaar ontmoeten in wat een stekker genoemd wordt, komen de twee bij elkaar en vormen electriciteit.
- 6 Electriciteit kan in batterijen worden opgeslagen. Grote batterijen bevatten niet altijd meer dan kleine. In grote batterijen wordt de electriciteit met een schop aangebracht, in kleine transistorbatterijen met een pincet.
- 7 Niet nieuwsgierige mensen draaien aan een schakelaar en denken er verder nooit bij na. Ik heb daar nooit genoeg mee genomen. Ik wil weten wat erachter steekt. Daarom kan ik u het volgende vertellen: Een lichtschakelaar is zoiets als een klem die de draad zo hard dichtknijpt dat de electriciteit er niet doorheen kan. Als de schakelaar wordt bediend, verlicht men deze druk en de electriciteit stroomt naar de lamp waar een stukje gloei-draad een soort strip-tease uitvoert. In elk geval is de draad daar naakt. Hier zien we dan ook de pure electriciteit in de vorm van een vonk, welke versterkt wordt door het bol-vormige glas, dat als vergrootglas werkt.
- 8 Waarom hebben lampen een beperkte levensduur? Omdat de hitte de aanwezige zuurstof in water verandert. Op den duur raakt de lamp vol water en dooft uit.
- 9 Het heeft me altijd verbaasd dat de industrie, die zo vooruitstrevend is op alle gebieden (uitvinding van gekleurde electriciteit voor tv en verkeerslichten, uitvinding van negatieve electriciteit voor koelkasten) toch 200 jaar nadat James Watt de elektrische ketel uitvond, nog steeds te dun verzekeringsdraad maakt.
- 10 De industrie maakt zeker goed stevig draad dat onder andere in de kippe-industrie (gaas-vormig) wordt gebruikt. Dat is beter dan de rommel die de electriciens verkopen, hoewel die ook moeten leven.
- 11 Ik heb nu een zekeringkast vol kippegaas, die al 6 maanden lang geen enkele moeilijkheid meer oplevert. Zelfs niet toen de meter uit elkaar barstte omdat de centrale er teveel electriciteit in stopte.
- 12 Waarom zekeringen? Een transistor die je afzekert beschermt zijn zekering door altijd als eerste de geest te geven.
- 13 Ik zit nog met de automatische toaster. Pop, een gaar toastje. Hoe weet die electriciteit dat ie gaar is? Nou ja, je kunt niet alles weten.
- 14 Ik zou nog enkele logische antwoorden kunnen geven, maar... Het licht gaat uit. Zeker een haspel in de centrale leeggelopen.

Bron: hack-tic 8, 1990

Van de bestuurstafel

Deze tijd van het jaar kun je beter bestuurslid zijn van een organisatie die zich bezig houdt in de recreatieve sector. Zwembaden, ijsverkoop en dergelijke. Dan krijg je als bestuurslid misschien nog de mogelijkheid om wat afkoeling te zoeken. Op 23 mei vertrokken in alle vroegte het bestuur en een aantal leden richting Almere-Stad. Je weet wel dat plaatsje dat is opgetrokken waar vroeger de Zuiderzee was. Helaas, helaas we waren veel te laat. De Zuiderzee was al gedempt.

Gelukkig was het in het zaaltje nog een beetje uit te houden. Jan Derksen had door drukke werkzaamheden zijn spreekbeurt, over het gebruik van Novell netwerken, onvoldoende kunnen voorbereiden. Het meeste heeft hij uit zijn geheugen moeten putten. Dat ging hem al met al niet slecht af. Uit de interessante vragen en opmerkingen kon je afleiden dat de toehoorders met belangstelling hebben zitten luisteren. Door die opmerkingen werd Jan soms ook geholpen om op het rechte spoor te blijven. Die hulp kon hij goed gebruiken. Het was voor hem uiteindelijk de eerste keer dat hij voor de leeuwen werd geworpen. En ook nog slecht voorbereid, dan kun je elk beetje steun goed gebruiken.

In gesprekken die ik nadien met een aantal van de aanwezige leden heb gehad kon ik opmaken dat er zeker op een vervolg gerekend kan worden. Over de technische kant van het gebruik van netwerken valt ook nog wel een en ander te vertellen. Nou dat zullen we dan maar te goed houden.

Waar ik als voorzitter het meest waardering voor heb gekregen is voor de toehoorders. Geen wanclank, in tegendeel op een positieve manier is de spreker een handje geholpen. Waarom ik dit nog een keer wil benadrukken is eigenlijk omdat ik jullie wil laten weten dat je niet bang hoeft te zijn om af te gaan tegenover je medeleden. Met andere woorden, trek ook eens de stoute schoenen aan. Kom ook eens op een van de bijeenkomsten een voordracht houden. Toch nog een beetje benauwd? Maar je denkt wel dat je een interessant onderwerp hebt? Neem eens contact op met een van de bestuursleden. Zij zijn bereid om je te helpen. Niet langer twifelen, gewoon doen.

Tijdens de bestuursvergadering ging het er ook warm aan toe. Nou niet meteen beginnen te gniffelen en denken dat we de grootste ruzie hebben gehad. Het was die avond buiten gewoon warmer dan binnen. Tijdens het vergaderen komt er wel eens een onderwerp ter tafel die een pittige discussie doet losbranden. Tja, en dan duurt het niet lang, dan wordt het binnen net zo warm als buiten. Gelukkig zaten we ook bij Jacques op een goed adres en werden de gemoederen regelmatig afgekoeld met een lekker glas koud drinken. Als je elkaar dan na zo'n afkoeling weer eens aan zit te kijken kunnen we meestal een glimlach niet onderdrukken. We gaan dan ook weer lachend en vol goede moed verder. We willen tenslotte allemaal zo snel mogelijk naar het einde van de agenda. Zonder dat we ons het kaas van het brood laten eten.

Almere-Stad. Je weet wel dat plaatsje dat is opgetrokken waar vroeger de Zuiderzee was. Helaas, helaas we waren veel te laat. De Zuiderzee was al gedempt.

Waarom vertelt hij ons dat allemaal zullen jullie denken. Wel er heeft in ons lijfblad een aantal malen een vacature gestaan voor een P.R.-functionaris, helaas heeft er tot nu toe niemand op gereageerd. Het bestuur vraagt zich af waarom. De werkzaamheden in het bestuur en de specifieke werkzaamheden voor die functionaris zijn toch best aardig om te doen. Je kunt er een hoop ervaring mee opdoen en leuke contacten leggen voor de toekomst.

Van achter de tafel wil ik dan ook iedereen oproepen nog eens goed over deze functie na te denken. Maar laat het niet alleen bij denken als je er niet uitkomt schroom dan niet om ons te bellen of te schrijven, dan kunnen we misschien met je mee denken. Dat lukt ook zonder aandelen NMB.

De volgende clubbijeenkomst wordt gehouden in Almelo. De reden dat we van de vaste volgorde afwijken komt omdat we van mening zijn dat de laatste bijeenkomst, waarop er weer bestuursverkiezingen zijn, eens een keer in een ander deel van het land moet worden gehouden.

Tot ziens dus op 19 september a.s. in Almelo.

Uw lij(e)der, Tonny Schäffer

Informatie

De μ P Kenner (De microprocessor Kenner) is een uitgave van de KIM gebruikersclub Nederland. Deze vereniging is volledig onafhankelijk, is statutair opgericht op 22 juni 1978 en ingeschreven bij de Kamer van Koophandel en Fabrieken voor Hollands Noorderkwartier te Alkmaar, onder nummer 634305. Het gironummer van de vereniging is 3757649.

De doelstellingen van de vereniging zijn sinds 1 januari 1989 als volgt geformuleerd:

- Het vergaren en verspreiden van kennis over componenten van microcomputers, de microcomputers zelf en de bijbehorende systeemsoftware.
- Het stimuleren en ondersteunen van het gebruik van micro-computers in de meer technische toepassingen.

Om deze doelstellingen zo goed mogelijk in te vullen, wordt onder andere 5 maal per jaar de μ P Kenner uitgegeven. Verder worden er door het bestuur per jaar 5 landelijke bijeenkomsten georganiseerd, beheert het bestuur een Bulletin Board en wordt er een softwarebibliotheek en een technisch forum voor dediverse systemen in stand gehouden.

Landelijke bijeenkomsten:

Deze worden gehouden op bij voorkeur de derde zaterdag van de maanden januari, maart, mei, september en november. De exacte plaats en datum worden steeds in de μ P Kenner bekend gemaakt in de rubriek Uitnodiging.

Bulletin Board:

Voor het uitwisselen van mededelingen, het stellen en beantwoorden van vragen en de verspreiding van software wordt er door de vereniging een Bulletin Board beschikbaar gesteld.

Het telefoonnummer is: 053-328506 of 053-303902.

Software Bibliotheek en Technisch Forum:

Voor het beheer van de Software Bibliotheek en technische ondersteuning streeft het bestuur ernaar zgn. systeemcoördinatoren te benoemen. Van tijd tot tijd zal in de μ P Kenner een overzicht gepubliceerd worden. Dit overzicht staat ook op het Bulletin Board.

Correspondentie adres

Alle correspondentie betreffende verenigingszaken kan gestuurd worden aan:

KIM Gebruikersclub Nederland
Postbus 1336
7500 BH Enschede

Het Bestuur

Het bestuur van de vereniging wordt gevormd door een dagelijks bestuur bestaande uit een voorzitter, een secretaris en een penningmeester en een viertal gewone leden.

Tonny Schäffer (voorzitter)
Paul Krügerstraat 27
7532 PW Enschede
Telefoon 053-613678

Jacques H.G.M. Banser (penningmeester)
Haaksbergerstraat 199
7513 EM Enschede
Telefoon 053-324137

Gert van Opbroek (secretaris)
Bateweg 60
2481 AN Woubrugge
Telefoon 01729-8636

Joost Voorhaar (redactie μ P Kenner)
Jekerstraat 67
7523 VP Enschede
Telefoon 053-333483

Jan D.J. Derksen (DOS65 coördinator)
Verfaillweg 434
1783 BP Den Helder
Telefoon 02230-28168

Geert Stappers (KGN/68k coördinator)
Engelseweg 7
5825 BT Overloon
Telefoon 04781-41279

Ereleden:

Naast het bestuur zijn er een aantal ereleden, die zich in het verleden bijzonder verdienstelijk voor de club hebben gemaakt:

Erevoorzitter:
Siep de Vries

Ereleden:
Mevr. H. de Vries-van der Winden
Anton Müller
Rinus Vleesch Dubois

